Available online www.ejaet.com

European Journal of Advances in Engineering and Technology, 2024, 11(2):27-34

Review Article **ISSN: 2394 - 658X**

Exploring Research Frontiers: MBSE and Requirements Engineering Integration Challenges

Iqtiar Md Siddique

Department of Industrial, Manufacturing & Systems Engineering, the University of Texas at El Paso, US.

Email: iqtiar.siddique@gmail.com

ABSTRACT

This research investigates the integration challenges at the forefront of merging Model-Based Systems Engineering (MBSE) with Requirements Engineering. It examines the complexities and nuances involved in seamlessly combining these two domains to enhance system development processes. Drawing from current literature and industry practices, the study identifies key obstacles and explores potential solutions for bridging the gap between MBSE and Requirements Engineering. Through a comprehensive analysis of research frontiers, the paper aims to shed light on the intricate interplay between these disciplines and offer insights into overcoming integration challenges. The findings from this investigation provide valuable guidance for practitioners and researchers seeking to navigate the evolving landscape of systems engineering. Ultimately, the study contributes to advancing the understanding and implementation of effective integration strategies between MBSE and Requirements Engineering in the context of modern system development initiatives.

Key words: MBSE

1. INTRODUCTION

Modeling and simulations have always been crucial in engineering, especially in the development of complex systems like land vehicles. Traditionally, wind tunnels were employed to assess aerodynamic performance using scale and full-sized models. However, with advancements in computing power, virtual models in virtual wind tunnels have become prevalent. These virtual approaches offer cost-saving benefits by eliminating the need for expensive physical models and facilities. Moreover, they enable the exploration of innovative design possibilities through recursive algorithms that iteratively refine vehicle shapes to achieve desired aerodynamic properties. This pattern of physical modeling augmented by computer-aided design, modeling, simulation, and manufacturing repeats at lower system levels. In software engineering, tools like the Unified Modeling Language (UML) and the Systems Modeling Language (SysML) are utilized to visualize, specify, and document complex systems, spanning hardware, software, personnel, and facilities. Recently, there's been a surge in interest in Model-based Systems Engineering (MBSE), which integrates formal modeling across the entire system lifecycle. MBSE, as defined by the International Council on Systems Engineering (INCOSE), involves applying modeling to support various system activities from conceptual design to development and later lifecycle phases. This expansion of formal modeling, particularly into the conceptual design phase, marks a significant advancement. It aims to address historical challenges such as poor requirements management, configuration management, and communication issues between stakeholders. Advocates of MBSE argue that it offers several benefits, including improved requirements elicitation and management, enhanced configuration management, and better communication among responsible parties throughout the system lifecycle [3].

While Model-based Systems Engineering (MBSE) promises numerous benefits, its application requires careful consideration. Models serve as abstractions of real-world domains, with modelers selecting parameters to focus on specific aspects. However, it's impossible to capture all real-world parameters due to limitations in identification and understanding of interactions. Models vary in their level of abstraction, depending on their intended purpose—some focus on specific aspects, while others encompass the entire domain with abstracted parameters. Executable models result in simulations, introducing complexities related to managing time-based parameters. Despite the potential of modeling and simulation in supporting systems engineering, it's crucial to exercise caution. All models are simplifications of reality, and while some are useful, they are inherently inaccurate to some degree. Therefore, while leveraging modeling and simulation, it's essential to recognize their limitations and potential inaccuracies.[2]. This study examines and organizes Model-Based Systems Engineering (MBSE) and Multidisciplinary Design Analysis and Optimization (MDAO) approaches to enhance comprehension of their association within systems engineering projects. Insights gathered from this literature review will inform the framework of the French project Concorde. A key objective of the project is to develop and execute a methodology for integrating components of the MDAO modeling approach directly from the MBSE framework, focusing on a case study involving Unmanned Aerial Vehicles (UAVs) [7]. firmly believe that in the coming decade, MBSE will emerge as a foundational paradigm for the development of sophisticated 21st-century complex systems and will play a vital role in facilitating effective collaborative development environments. The primary hurdle will be to guarantee that the system model accurately represents the perspectives and requirements of stakeholders, serving as a unified working platform where all stakeholders can contribute and collaborate. The aim is to ensure that the resulting system aligns with stakeholders' expectations and fulfills their envisioned outcomes [8]. This paper examines requirements management practices within rail transport projects, particularly during the planning and acquisition phases. Linear rail networks consist of interconnected cyber-physical systems. With the transportation sector increasingly adopting digital twinning to enhance asset lifecycle management strategies, requirements management functions are tasked with managing a complex array of asset information requirements [9]. Some researchers underscore the significance of digital transformation as a central strategic endeavor aligned with the U.S. Department of Defense Digital Engineering Strategy. Acknowledging the imperative to enhance the U.S. Navy's competitive edge, this paper outlines the fundamental elements necessary to modernize existing engineering design and development processes [10]. Henry et al. (2012) introduces a universal SysML-based approach aimed at enhancing the Architectural Design phase of Systems and overcoming obstacles to MBSE adoption. This proposed methodology offers an alternative approach to formalizing requirements, delineating design alternatives, automating design space exploration, and evaluating optimization outcomes within the native modeling framework. A concrete example is presented to illustrate the application of this methodology. Additionally, a glossary is appended to the paper for reference [11].

A model's effectiveness is determined by its suitability for a specific purpose, and it can only serve the intended purpose for which it's designed. Attempting to apply a model beyond its intended scope can lead to misinterpretation or even hazardous outcomes. Moreover, the development of models and simulations often presents its own set of challenges, and it's essential to avoid creating overly complex model-based processes that mirror the complexity of the problem they seek to address. Within the realm of Model-based Systems Engineering (MBSE), models can be applied to various systems engineering processes. In this context, we focus on how MBSE can support requirements engineering. We propose a framework to evaluate the effectiveness of MBSE in this domain, outlining the key activities involved in requirements engineering and assessing how MBSE can enhance each of these activities. Additionally, we highlight several challenges that need to be addressed before MBSE can be effectively applied to requirements engineering.

2. APPLICATION OF MBSE TO REQUIREMENTS ENGINEERING

Model-based systems engineering (MBSE) is a structured approach that supports various aspects of complex system development, including requirements, design, analysis, verification, and validation. Compared to traditional document-based systems engineering, MBSE in a digital modeling environment offers several advantages, leading to its increased adoption. These advantages include cost savings through reduced development time and improved software security. The Software Engineering Institute (SEI) CERT Division is exploring how MBSE can also help mitigate security risks early in the system development process, ensuring systems are secure by design rather than adding security features later. While MBSE does not prescribe a specific process, any MBSE approach should encompass four systems engineering domains:

requirements/capabilities, behavior, architecture/structure, and verification and validation. This post focuses on how MBSE addresses the requirements domain, which outlines the problems the system aims to solve [5].

Traditionally, requirements serve as the starting point for any project, system, or product change. They can take various forms, such as high-level business goals, mid-level user needs, or detailed system capabilities and behaviors. Before being incorporated into an MBSE model, requirements typically undergo classification, deduplication, and rephrasing. Within an MBSE model, requirements can be classified into three main types:

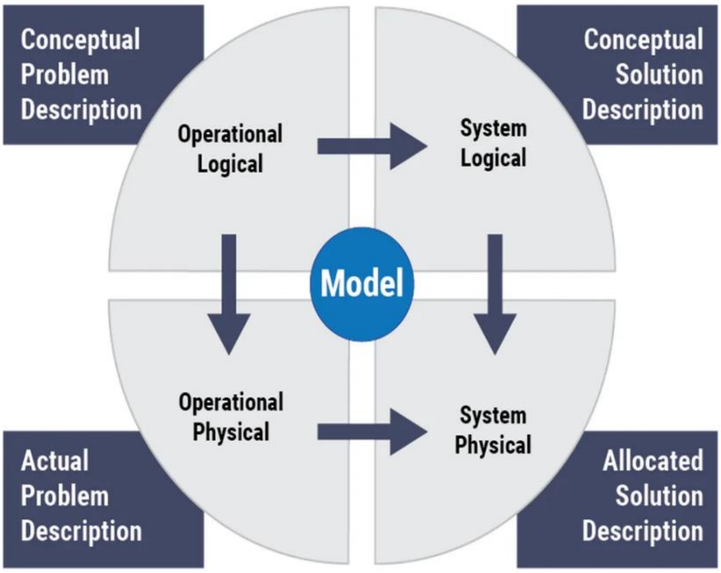

Figure 1: Model Components [5]

Business requirements: These are high-level statements outlining the goals, objectives, or needs of an organization. They typically identify opportunities or challenges relevant to the organization's mission.

User requirements: These statements represent the needs of specific stakeholders or stakeholder groups. They describe how individuals or groups expect to interact with the intended solution, bridging the gap between high-level business requirements and detailed solution requirements.

System requirements: These detailed statements specify the capabilities, behaviors, and information required for the solution. They include conditions under which the solution must remain effective, as well as non-functional requirements such as security, usability, and modifiability.

By utilizing MBSE to manage requirements, organizations can streamline the development process, improve communication, and ensure alignment between stakeholder needs and system capabilities.

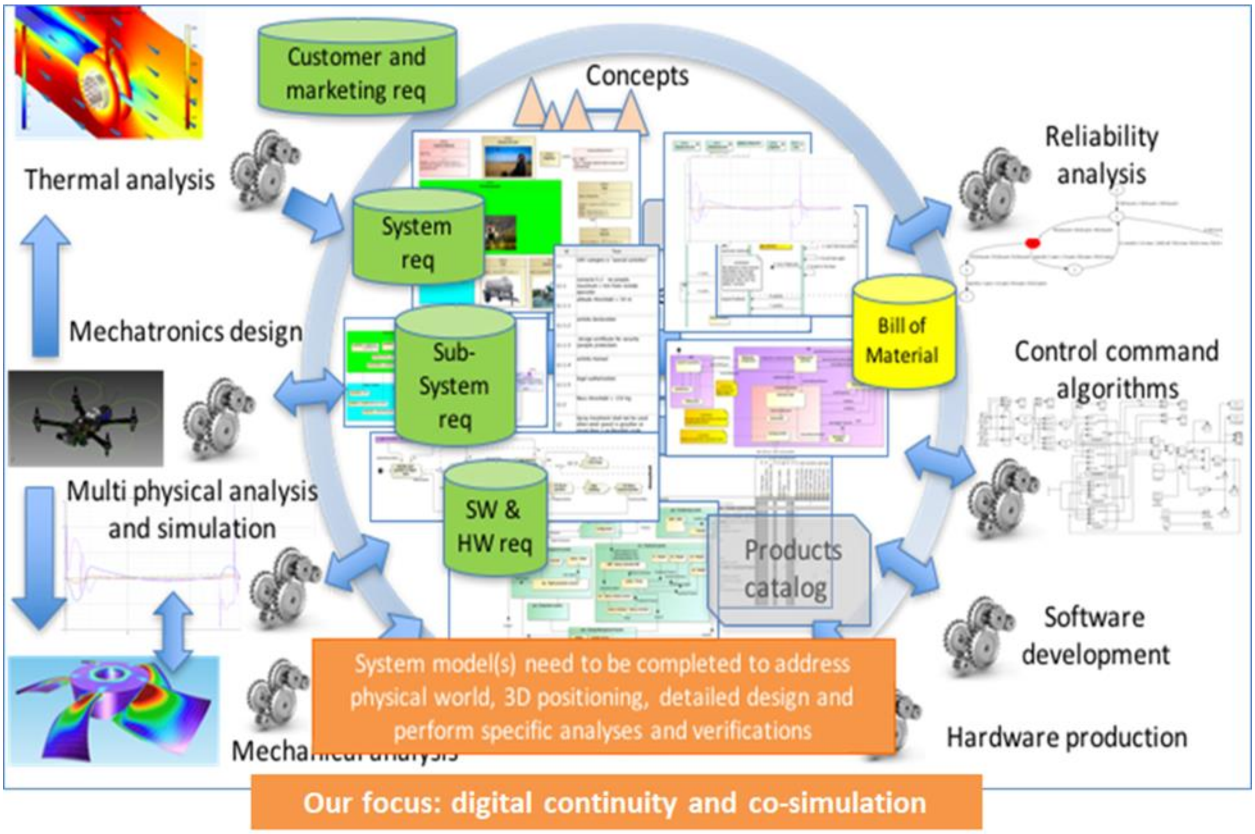

Figure 2: Model Based System Engineering

- **MBSE Support for Requirements Elicitation/Generation**

Requirements elicitation and generation involve collaborating with customers and end users to understand the problem at hand, identify functional and non-functional requirements, specify system performance criteria, and outline any constraints. Some people Explores the challenges presented when Model-Based Systems Engineering (MBSE) is approached solely as a modeling activity, lacking full integration with advanced simulation capabilities. It also delves into the potential for various forms of closer integration between these two streams. Using an example of an unmanned vehicle fleet offering emergency ambulance services, it analyzes the need for a comprehensive Modeling and Simulation (M&S) methodology to effectively address such complex systems [6]. Model-based Systems Engineering (MBSE) could enhance requirements elicitation and generation through the following means:

Facilitating the capture of requirements in suitable formats.
Assisting in the creation, exploration, and validation of use cases or operational scenarios.
Providing support for various requirements elicitation techniques such as workshops and interviews.
Enabling simulation and visualization of stakeholders' operational environments.
Supporting functional allocation and grouping processes.
Facilitating feasibility analyses and trade studies.
Allowing for abstraction of system information at any desired level.

- **MBSE Support for Requirements Analysis and Negotiation**

Requirements collected from users undergo analysis to ensure a comprehensive understanding and undergo negotiation to reach consensus among stakeholders on well-defined requirements. Requirements Analysis aims to identify overlapping, inconsistent, infeasible, or excessively costly requirements, striving to improve the overall system solution through refinement. Model-based Systems Engineering (MBSE) can aid requirements analysis and negotiation through the following means:

Facilitating the establishment of quantitative performance criteria for the system.

Providing context for understanding the system within its operational environment.
Assisting in deriving requirements related to human-system interfaces and aesthetic considerations.
Supporting functional elaboration by decomposing higher-level requirements into lower-level ones, ensuring a flow-down of requirements.
Enabling the tagging of requirements with additional information such as identifiers, titles, priorities, criticality, feasibility, risks, sources, types, rationales, histories, and interrelationships.
Supporting the analysis of requirement subsets to identify areas requiring further investigation and negotiation with stakeholders.
Facilitating trade studies to explore different design options.
Enabling the derivation of design constraints.
Existing literature suggests that all computer-aided requirements analysis and validation tools rely on model-based approaches grounded in formal knowledge representation and artificial reasoning.
(Reubenstein & Waters 1991, Rich & Waters 1998, Pohl et al 1994, HCI 2005, Sutcliffe et al 1998, James 2000, Maiden 1998, Scott & Cook 2008).

- **MBSE Support for Requirements Allocation Based on functional and system analysis:**

Requirements need to be allocated to functional and physical groupings within the system architecture. Model-based Systems Engineering (MBSE) can assist in requirements allocation through the following means:
Facilitating comparisons of candidate system functional architectures to determine the most suitable functional groupings that meet stakeholder requirements.
Supporting evaluations of candidate physical architectures to translate functional requirements into physical components effectively.
Assisting in tender evaluations if the synthesis occurs through tendering processes.
For requirements validation, it's essential to ensure completeness, consistency, and clarity. MBSE can aid in requirements validation by:
Supporting validation of individual requirements to ensure they meet specified criteria.
Assisting in validating the entire set of requirements to ensure coherence and comprehensiveness.
Facilitating planning and execution of various tests like Developmental Testing and Evaluation (DT&E), Acceptance Testing and Evaluation (AT&E), and Operational Testing and Evaluation (OT&E).
Requirements management, or configuration management, involves maintaining and justifying requirements throughout the system lifecycle. MBSE can support requirements management by:
Ensuring compliance with specified requirements engineering processes.
Enforcing structure, syntax, and semantics of requirements.
Facilitating reviews by providing baselines and enabling traceability between different versions.
Supporting change management processes including identification, analysis, approval, and implementation of changes.
Assisting in generating reports related to requirements management activities.

- **Utilizing MBSE for Requirements Engineering Support**

To fully leverage the potential of Model-based Systems Engineering (MBSE) in system development processes, several critical issues must be addressed through further research. While the INCOSE Systems Engineering Vision 2020 identifies inhibitors such as challenges in integrating models across organizational and lifecycle boundaries, limitations in model/data exchange capabilities within modeling tools, and a shortage of MBSE skills, these are just a few concerns among many. The widespread adoption of MBSE faces numerous unresolved issues beyond those mentioned by INCOSE. This paper aims to outline these issues rather than solve them comprehensively. It highlights that many MBSE initiatives are relatively simplistic in addressing the underlying complexities that currently hinder MBSE from realizing the ambitious vision outlined by INCOSE.

- **Modelling**

Precisely and unambiguously representing stakeholders' real-world concerns in a model presents two significant challenges: The first challenge lies in effectively modeling the real world, especially its soft variables. Despite the availability of modeling techniques like system dynamics since the 1970s, their adoption has been limited, even in environments where they could offer substantial benefits. Emerging approaches like agent-based modeling hold promise, but their adoption is hindered by stakeholders' limited awareness of enterprise modeling capabilities and associated benefits. Overcoming this challenge is crucial for broader adoption of Model-based Systems Engineering (MBSE). The second difficulty involves integrating a wide range of stakeholder perspectives into a single model to create a valid representation of the system. While stakeholders can be categorized into a few groups, merging their views into a cohesive model remains challenging. Existing methodologies for collecting, interpreting, negotiating, and managing these collective views are scarce. This challenge extends beyond MBSE and is a persistent issue in systems engineering and requirements engineering, especially at the enterprise level. For instance, defense architecture frameworks require developing over sixty views, each demanding significant investment and coordination efforts. The growing volume of information risks diluting holistic perspectives, prompting a reassessment of the value of comprehensive architecture views in influencing systems of systems development outcomes. (Dahmann and Baldwin 2011) [1].

- **Database management**

Significant effort in data administration is imperative for the successful integration of models both horizontally and vertically. Without standardized schemas, common data structures, standard data definitions (a data dictionary), and agreed data translations (a thesaurus), Model-based Systems Engineering (MBSE) initiatives are likely to encounter obstacles. These issues, reminiscent of long-standing challenges in software development, require straightforward solutions such as explicit definitions of model elements like "operator" or "maintainer" in terms of skill sets and training. If models are to be shared across organizations, these definitions must be standardized, or a shared thesaurus must facilitate translation between different data elements. However, despite the simplicity of the solution, many organizations lack the resources for comprehensive data administration, and projects often neglect to subscribe to standard definitions. Furthermore, when multiple organizations are involved, the issue of common standards arises, along with the challenge of standardizing standards among various standards organizations. Security considerations are paramount in system design and operation, necessitating the classification of information into multiple security levels or at least privacy levels. The system must support access rights to ensure traceability, with customization options for views based on stakeholders' clearance levels and need-to-know requirements. Mechanisms for sanitizing data and exporting subsets to different users are essential, albeit posing significant configuration management challenges. To support various levels of access and data subset capabilities, data must be tagged in multiple ways to implement diverse business rules. This complexity extends to classifying models, diagrams, paragraphs, and sections according to the highest level of classification of constituent elements. Standards are crucial to establishing common views of access rights among nations, contractors, and customers, ensuring security and confidentiality in MBSE endeavors.

- **Lawful Context**

The current legal framework predominantly relies on documents to formalize legal agreements, posing a challenge for Model-based Systems Engineering (MBSE), which operates primarily through models. However, there are examples demonstrating how models can be translated into document form to comply with this paradigm. To fully unlock the potential of MBSE, there must be a framework for legally recognizing baseline models in binding contracts. Achieving this requires resolving issues related to data administration, security, data ownership, and change and access management through standardized solutions, although this goal is still some distance away.

Moreover, MBSE initiatives will face obstacles unless appropriate mechanisms are established to manage intellectual property (IP) and address complex issues surrounding data, model, and tool ownership. For the MBSE vision to materialize, each customer would need exclusive rights to utilize and transfer any model developed under a contract to future contractors or subcontractors. However, this practice has not been common, as contractors are typically reluctant to share models of their proprietary systems with competitors. Similarly, few customers are willing to allow others to use their proprietary models. These commercial considerations are unlikely to change in the foreseeable future [4].

- **Execution of Authority**

The process depicted in Figure 1 is inherently dynamic, as changes to one or more elements of the models (such as operational, business, or stakeholder requirements) are likely to occur frequently. These changes must be carefully managed, especially when they need to flow through from one model to another, for example, from a change in a personnel model to the subsystem model accommodating that aspect of the system. In cases where fixed-price contracts are in place between the business model owner and the subsystem model owner, extensive configuration management is necessary to ensure that changes are made in a controlled manner and are synchronized with other efforts such as contract administration. The process must also address situations where funding constraints prevent changes, leading to discrepancies between the subsystem model and the real world.

Key issues in change management include:

Control over changes to any model or data, with permission required from a change control board.

Saving model elements with changes under new names to create new baselines, with the ability to roll forward and back changes in the database.

Recording the owner of model elements or data to obtain their permission for changes affecting their data.

Locking certain model elements to restrict changes to specific parties, for example, preventing contractors from modifying the customer's operational model.

Simplifying visualization of change effects for communication with non-technical stakeholders.

Supporting both informal "what if" investigations and formal change management processes.

Ensuring that the customer changes to an operational model during the contract either flows down into the contract through subsequent amendment processes or retains the version number of the original model for system testing.

Other critical issues include:

Utility: Ensuring commonality of models and tools within and across organizations.

Management: Ability to manage baselines, contractual boundaries, and model partitioning while maintaining version control.

Stakeholder visualization: Providing standard visual interfaces for stakeholders to easily understand their business in any model.

Flexibility: Allowing for a mixture of computer-based and legacy tools and models, as well as paper-based requirement sets.

Trust: Establishing mechanisms to ensure the integrity of data, models, and tools.

Validation: Continuously validating models and interfaces to ensure accuracy and reliability.

Interfaces with other models and tools: Enabling seamless integration between SE models and other systems such as project management tools and databases.

Effectiveness: Demonstrating cost-effectiveness and improvement in project outcomes using MBSE tools and practices.

As these issues are addressed, tools become more sophisticated and widely used, and evidence of the effectiveness of MBSE accumulates, the case for integrated MBSE will strengthen, potentially leading to widespread adoption.

3. CONCLUSION

In conclusion, this study has explored the intricate landscape of integrating Model-Based Systems Engineering (MBSE) with Requirements Engineering, shedding light on the challenges and opportunities at the forefront of this endeavor. Through an examination of current research frontiers, we have identified key obstacles hindering seamless integration and proposed potential solutions to address these challenges. It is evident that while there is significant potential for synergy between MBSE and Requirements Engineering, achieving effective integration requires careful consideration of various factors such as tool interoperability, data management, and stakeholder collaboration. Moreover, the evolving nature of systems engineering practices and the increasing complexity of modern systems pose additional challenges that must be navigated. Moving forward, it is imperative for practitioners and researchers to continue exploring innovative approaches and methodologies for integrating MBSE and Requirements Engineering. Collaboration across disciplines, active engagement with industry stakeholders, and ongoing knowledge sharing will be crucial in advancing the state-of-the-art in this domain. By overcoming integration challenges and leveraging the complementary strengths of MBSE and Requirements

Engineering, organizations can streamline the system development process, enhance communication and collaboration, and ultimately deliver higher-quality, more robust systems that meet stakeholder needs and expectations.

REFERENCES

[1]. Dahmann J. and Baldwin K. "Implications of Systems of Systems on Systems Design and Engineering", Proceedings of the 2011 6th International Conference on Systems of Systems Engineering, Albuquerque, New Mexico, 2011.

[2]. Box, G.E.P, and Draper, N.R., Empirical Model-building and Response Surfaces, Wiley, 1987.

[3]. Friedenthal, S., Griego, R., and Sampson, M., "INCOSE Model based Systems Engineering Initiative", presentation to 17th Annual INCOSE International Symposium, San Diego, 2007. Quoted in C. Haskins, "A Historical Perspective of MBSE with a View to the Future", INCOSE International Symposium, Denver, 2011.

[4]. Do Q, Cook S, Campbell P, Scott W, Robinson K, Power W, & Tramoundanis D. "Requirements for a Metamodel to Facilitate Knowledge Sharing between Project Stakeholders", Proceedings of the 10th Annual Conference on Systems Engineering 2012.

[5]. Shevchenko, N. (2022, November 28). Modeling Capabilities with Model-Based Systems Engineering (MBSE). Retrieved May 9, 2024, from https://doi.org/10.58012/5mgc-q360.

[6]. Zeigler, B. P., Mittal, S., & Traore, M. K. (2018). MBSE with/out Simulation: State of the Art and Way Forward. Systems, 6(4), 40.

[7]. Chaudemar, J. C., & de Saqui-Sannes, P. (2021, April). Mbse and mdao for early validation of design decisions: a bibliography survey. In 2021 IEEE International Systems Conference (SysCon) (pp. 1-8). IEEE.

[8]. Ramos, A. L., Ferreira, J. V., & Barceló, J. (2011). Model-based systems engineering: An emerging approach for modern systems. IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews), 42(1), 101-111.

[9]. Chen, Y., & Jupp, J. R. (2023). Challenges to requirements management in complex rail transport projects. International Journal of Product Lifecycle Management, 15(2), 139-177.

[10]. Voth, J. M., & Sturtevant, G. H. (2022). Digital engineering: expanding the advantage. Journal of Marine Engineering & Technology, 21(6), 355-363.

[11]. Broodney, H., Dotan, D., Greenberg, L., & Masin, M. (2012, July). 1.6. 2 Generic Approach for Systems Design Optimization in MBSE 1. In INCOSE International Symposium (Vol. 22, No. 1, pp. 184-200).