Systems Engineering Approaches for the Design of Complex Systems

Vasudha Srikumar

Abstract: The main focus of this survey paper is to provide various approaches for the design mechanism of complex systems and portraits Systems Engineering methods and way of thinking about the environment, problem and solution to design complex systems. Dealing with the complex systems is not new for Systems Engineers. Reliability, Availability and Maintainability (RAM) fields are crucial for a complex Systems Engineering (SE). A literature survey on Model-Based System Design (MBSD) is carried out, constructed models that helps to understand specifications, operation and performance of system components are explored. Different MBSD methodologies are used to describe the systems component functionality and relationship between those components. However, designers are facing difficulties to use optimum MBSD models for design of complex systems. This survey paper deals with systems complexity and collate existing approaches to identify complexity in an environment and a solution space.

Keywords: Complex System (CS), Model-Based System Design (MBSD), Systems Engineering (SE), Systems Design (SD), SysML.

1. INTRODUCTION

Systems Engineering is an interdisciplinary field of engineering and management that focuses on how to design, integrate and manage complex systems over their life cycles. Complex System possess many interactive components which provides feedback mechanisms, so that the results of the outputs are not in proportion to the original inputs. A small action can have widespread effects across the system, which is also referred to as non-linearity. The SE perspective is based on Systems Thinking, also SE utilizes systems thinking principles to organize the considerations of costs and consequences of current knowledge and practice gaps. Complex Systems Engineering is a intersection of CS, systems thinking and SE [4, 5]. It is important to make a strong connection between CS and SE, for that the following questions need be addressed.

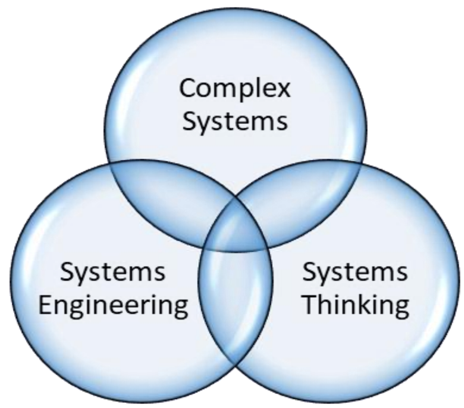

Figure 1: Complex Systems Engineering

1. What difficulties have CS encountered?
2. What analytical tools from the field of CS can be applied to SE issues?
3. What abilities have systems engineers improved to develop CS?

All these questions will be answered in a pragmatic way in this paper. The organization of this survey is as follows. Section 2 focus on systems thinking for complex systems, Section 3 provide a brief introduction to the practice RAM engineering and SE, importance of reliability, availability and maintainability, importance of integrating RAM through SE, Section 4 portraits the importance of MBSD in the perspective of design of CS and Section 5 explains logical approach to design CS along with conclusions and references.

2. SYSTEMS THINKING FOR CS

SE requires a change in thinking and an expanded set of analytical tools and methods to design CS. More number of principles are provided to encourage systems engineers to think diversely to deal with complexity. In this section, we will focus on the required changes in system thinking to acknowledge the complexity.

1. An important mechanism in the CS engineering is to observe and understand the CS patterns. This helps to avoid unnecessary prominence and side effects

and to provide the ease to add required system components and related interfaces.

2. The goal of a system should be met even if a number of current conditions of the change such as operational environment and so on. SE can take the maximum advantage of integrative thinking to generate improved solutions. Optimization is often counterproductive within a complex system [2]. Instead of optimizing, CS engineers should maintain balance among potential acquirers within the project
3. A combination of ability and humility improves the systems engineers innovation and fast learning to from continuous cycle of product of solutions in context. SE should develop the ability to acknowledge complexity, abandon control, new methods, and unexplored territory should be adopted. Systems engineers have to be determined to accept the unreliability, having an attitude of doubt of existing knowledge and to be open to learning from failure.
4. Able to adjust oneself to new or changed circumstances. SE should imitate how systems effectively deal with complexity. SE have to deal with CS by selecting the best versions, identifying and creating variations.
5. Develop gradually a specified solution to the problem instead of constructing a system from scratch, it is important to consider the complexity of the environment and the solution.
6. In architecting and designing, solutions can be confined to desired solutions. This is applicable when the system being designed must be able to withstand or recover quickly from difficult conditions.
7. Improving the collaborative mindset can lead to strong Stakeholder engagement practices to enable co-creation and co-operation in systems design. It includes active listening, information sharing, and making decisions transparent.
8. It is not possible to understand the CS with one method of analysis, SE should apply different methods iteratively. In depth analysis of the CS by addressing them at a higher or lower hierarchical level [3]. The CS engineer must be open to solutions that arise from the bottom-up rather than only seeking to impose order from the top-down.
9. Instead of focusing on detailed outcomes, focus should also be on what range of solutions can be adapted to get the desired results.
10. By understanding the multiple perspectives of CS, a SE can find creative ways to solve several problems at once.
11. Having mindset to learn from errors helps to mitigate the risk.
12. Evolving system with closely compacted elements problems and opportunities will be continually prominent.
13. Meta-cognition helps to identify and make useful patterns of thinking more frequent, and improve understanding of a complex situation.

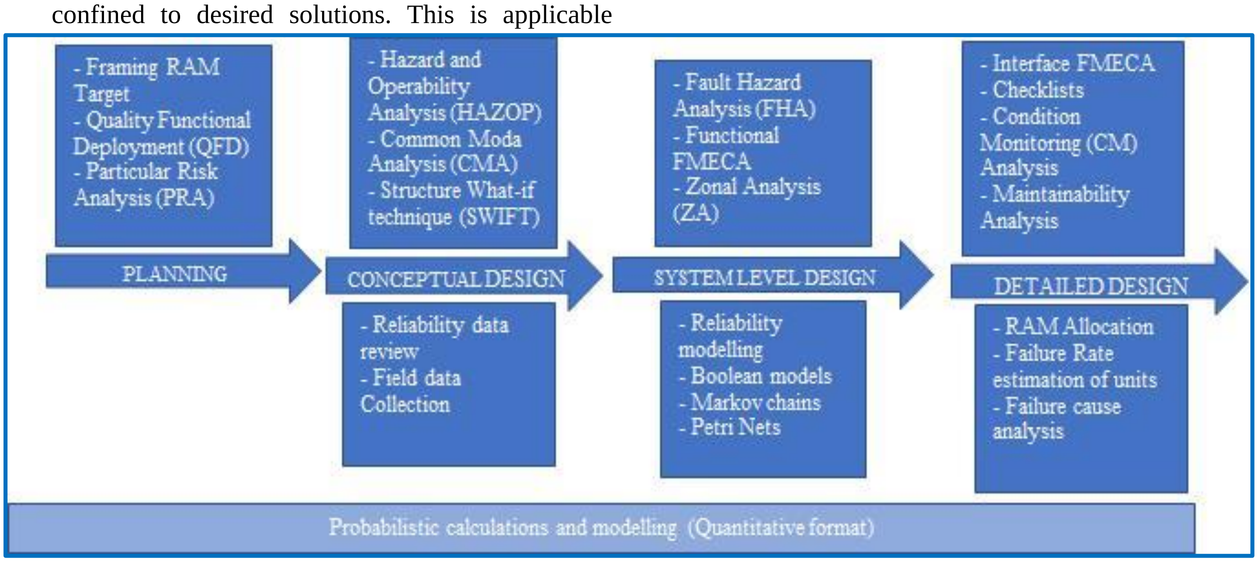

Figure 2: RAM techniques in SE process

3. IMPORTANCE OF RAM IN SE

Reliability, Availability and Maintainability (RAM) are system design attributes that have great impact on the support or maintenance of a developed system. The standard definition of reliability is the probability of zero failures over a defined time mission or interval. Availability is defined as the percentage of time a system is considered ready to use when tasked. Maintainability is a measure of the ease and rapidity with which a system or equipment can be restored to operational status following a failure. The RAM attributes can also impact the ability to perform the planned mission and decide overall mission's success. RAM engineering uses the SE knowledge and methods to handle the risk of failures and reducing uncertainties.

RAM analysis can be both qualitative and quantitative [1]. Qualitative analysis is the basis to quantitatively execute the systematic observation of the similarities or dissimilarities evaluation to support follow-up decision making. The purpose of qualitative analysis is to recognise failure modes, mechanisms and causes and determine the possible test strategies and maintenance. Figure 2 gives RAM analysis at various stages of a SE design process.

Systems concepts can be analysed with different point of view to capture the operational, functional, physical/architecture with the support of MBSE. SE can handle complexity for many engineering disciplines including RAM engineering. Model-based SE (MBSE) supports to design the system concepts using models. SysML is considered as the example SE tool for developing system architecture views. System Modelling Language (SysML) is a commonly accepted technology for MBSE, which uses the same profile as Unified Modelling Language (UML) along with SysML can support to SE activities like requirement allocation.

SE provides mechanisms to ensure the correctness of system concept and design team coordination. SE also includes analyses that can improve the basis on which the RAM analysis is carried out.

3.1. SE Activities for RAM Engineering

The following are the SE activities with a view on RAM integration.

1. Requirements Analysis
2. System Architecture Development and Analysis

3.1.1. Requirements Analysis

The SE activities start with elicitation of the requirements from stakeholders. Requirements analysis is the process of defining the expectations of the users for an application that is to be built or modified. It involves all the tasks that are conducted to identify the needs of different stakeholders. Requirements analysis involves activities such as to analyse, document, validate and manage system or software requirements. High-quality requirements of CS are documented, actionable, measurable, testable, traceable, helps to identify business opportunities and are defined to a facilitate system design. One of the important attribute of requirements is operational concept that provide an overview to describe operating environment, system missions, and the internal / external interfaces. The typical models used for capturing a conceptual architecture are operational context model, sequence diagram and use case diagram [1]. RAM engineers need to focus on specific required RAM performance under different operating conditions and RAM requirements shall contain environmental conditions, and operating conditions are specified.

3.1.2. System Architecture and Analysis in RAM Perspective

The system architecture and analysis takes the inputs from system requirements definition. RAM engineers examines each functions along with failures and describes as loss of function. Complex systems are better served by the SE suite of following tools to systematically develop a vision of behaviours, interfaces, elements and control structure [1].

3.1.2.1. Behavioral Analysis

In SysML, different types of functional models are categorized as shown in Figure 3.

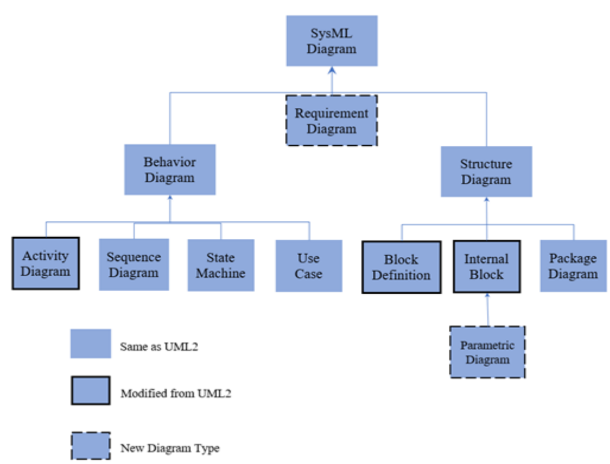

Figure 3: SysML models

Functional models in SysML are Activity Diagrams and State Transition diagrams. Activity diagrams represents the flow of operation of a process and State diagram is an event/trigger based model. State transitions can be triggered against the context captured in the Activity Diagram to ensure consistency. The state machine diagram explicitly describes the behaviour of system and it consists of different states and triggering events that drive the transition between states. For design of CS, only by depending on functional architecture to analyse RAM performance of could be incomplete, because functional architecture helps in identifying potential failure but not the associated cause. Therefore, the physical analysis of a system should be also be developed.

3.1.2.2. Physical Analysis

The physical analysis provides the components that become fully aware of the identified functions. For RAM analysis a technical system is usually taken from a functional point of view instead of architecture. The physical aspects of system is the physical decomposition like physical Failure Mode

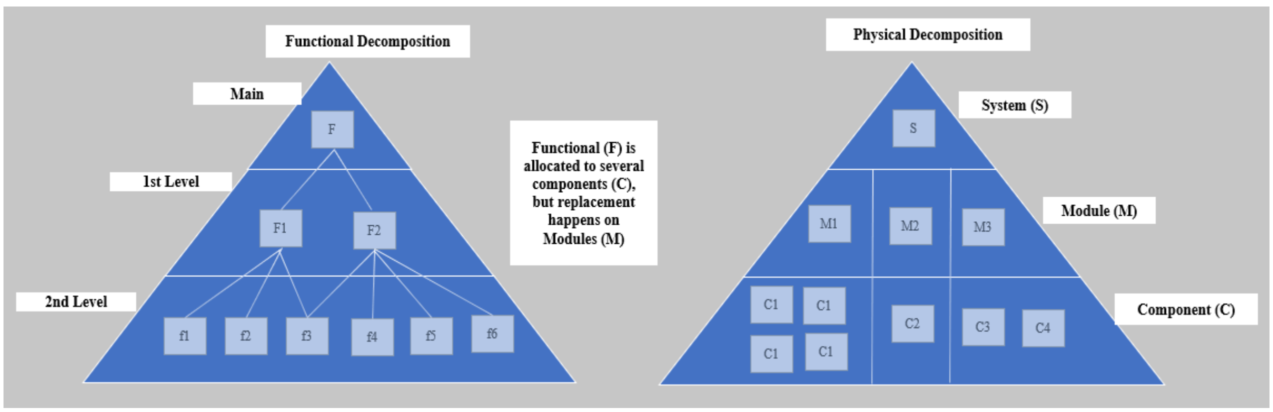

Figure 4: Modularity of design

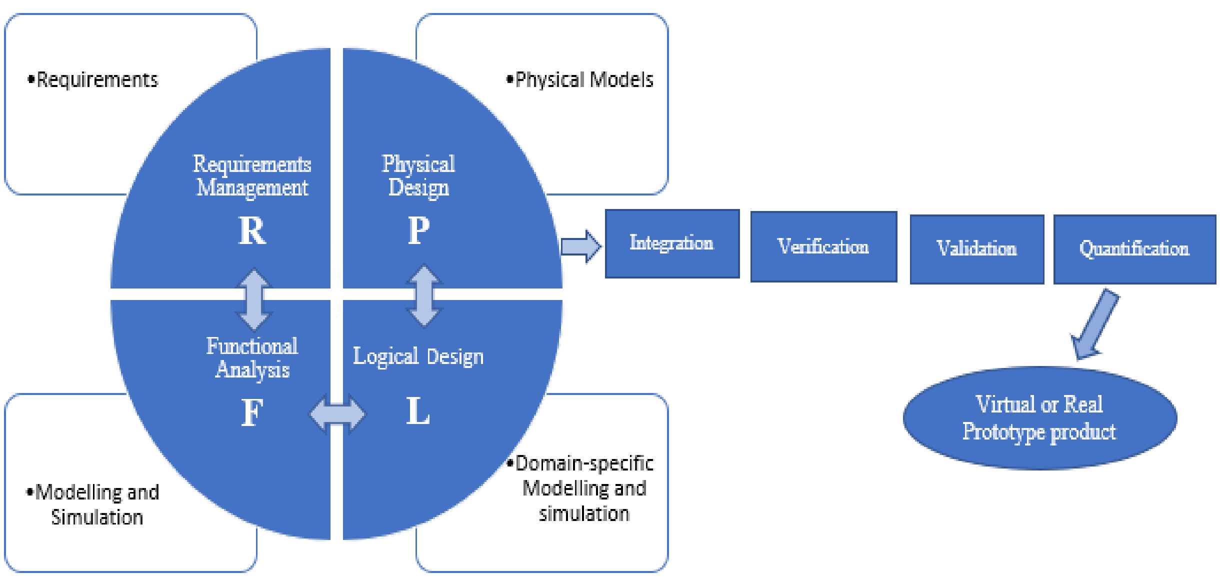

Figure 5: Iterative steps between requirements and simulation

Effects and Criticality Analysis (FMECA). RAM engineering for CS elements or parts are inter-related. Physical aspects in RAM study is a brainstorming process that requires participations from multiple stakeholders from different disciplines as shown in Figure 5. SysML models are able to develop block definitions that contain physical attributes such as weight and size and also ensure coverage and traceability of defined constraints. The complete physical analysis study provides the understanding of the local effects on basic components can disturb the system [8]. Some functions are explained by components located within different modules, but the replacement takes place at a module level. Figure 4 explains the design methods of functional and physical decomposition.

4. MODEL-BASED SYSTEMS DESIGN

Model-based design provides an efficient approach for establishing a common framework for communication throughout the design process while supporting the development cycle (Vee-Model). Models are created to deal with complexity, in doing so they allow us to understand an area of interest or concern and provide unambiguous communication amongst interested parties. Model-Based Systems Design (MBSD) provides modelling to support function analysis and requirement analysis. Systems Design (SD) plays an important role in large scale systems design. Usually SD process includes three iterative activities namely requirements analysis, function analysis and design synthesis. Following section tabulates different MBSD methods, respective pros and cons and supported tools, refer Table 1.

Major goals of MBSD include improved communication, improved quality, increased productivity, reduced risk, better awareness of the technical baseline and visibility of the design, focussed decisions on heritage design deviations [6]. The formalised application of modelling to support activities beginning from the conceptual design phase and continuing throughout the development and later lifecycle phases namely system requirements, analysis, design and V&V [7].

Table 1: MBSD methods, pros and cons

Methodology	Fundamental Idea	Language	Tool	Advantages	Limitations
Harmony SE5	Use Case-driven design	SysML	IBM Rational Rhapsody	a. User-friendly b. Seamless transition for subsequent V&V	Management of different abstract levels is tough, Insufficient requirement analysis capability, Lack of general criteria for system design
OPM67	Object process unified design	OPL	OPCAT (Object Process Case Tool)	a. Simple modeling elements to represent a complex system b. Effective decomposing capability for various system levels	Ability to represent specific information in detailed design stages is lacking, Requirements representation not supported, Lack of general criteria for system design
Vitech MBSE8	Concurrent design based on "onion model"	System definition language	Vitech's CORE	a. Integrated SD repository to coordinate design activities b. Object-oriented theory promoted as an example of MBSE best practice	Lack of general criteria for system design
OOSEM9	Object-oriented design	SysML	Tools that support SysML such as Magic Draw	a. Object-oriented theory b. Partitioning criteria is established to ensure a robust design	General criteria for good design beyond the partitioning criteria is not specified
SYSMOD10	SD based on "Vee" Model	SysML	SysML Plugin of Magic Draw	a. User-friendly b. Equipped with a detailed modeling process	Lack of general criteria for system design
SA11	Control architecture based on states	Varies (eg, UML/SysML, OWL2)	a. Eliminates inconsistencies between system engineering and software engineering b. Provides rigorous state-based foundation for SE	a. Eliminates inconsistencies between system engineering and software engineering. b. Provides rigorous state-based foundation for systems engineering.	Mainly used for control systems, Lack of general criteria for system design

5. APPROACHES TO ADDRESS SYSTEMS COMPLEXITY

This section focus on the particular SE methods to deal with complexity in problem, solution and environment contexts. The tools in each column generally range from the simplest at the top to the most complex at the bottom. The resulting matrix is shown in Table 2 and Table 3, provide appropriate modelling approaches to SE in order to address complexity in CS. Complexity is an attribute of the technical system being developed but also of the problem space (including people and organizations) and the environment. Complexity is associated with size, diversity, dynamism and with emergence. It is a challenge to systems engineers not to over-simplify in pursuit of representations and capabilities that can be understood and controlled; the right level of complexity is a key [2].

Table 2: SE Techniques to deal with the environmental issues

Complexity / Approaches	Requirements Inference	Trade Studies	Solution Architecture and Design	Development Process
Environmental Complexities - General	Requirements elicitation to be done appropriately followed with the aid of below points: a. Adequate System goals b. Multiple techniques c. Different perspectives d. Numerous data collection e. List down System goals	Focus on achieving a robust / sturdy system	Positive and Negative feedback mechanism to be considered	Software techniques to identify the nature of the problem (internal structure and information flows, and produce simple representations)
Complex and Self Regulatory Interchange with the Environment	Requirements to provide flexibility to control the system	Hierarchical approach to handle the issues and complexities of performance	Prototyping the Product at an earlier stage	Test Software design with its outputs or feedbacks available to the developers for analysis
Environment subjected to unpredictable events and/or Multiple complexities	Use of effective protocols to focus on the Requirements elicitation on resiliency, robustness and adaptiveness	Strength / durability of the System to be used as a key element to derive the System architecture	a. Robustness of the System b. Resilience to achieve "beyond-design-envelope" events for Functional state recovery	a. Resilience analysis. b. Case studies on how the other Systems are surviving the adversity
Complexity in the Operation	Identification of requirements constraints	Process oriented approach for apt trade studies	a. Adaptable and/or reconfigurable elements. b. Accomplish scenarios to achieve System objectives	a. Agile, evolutionary SE processes b. Multi-layer processes and their interface points
Complexity in Stakeholder Connection	Determine scalability of "goodness" or fit to use requirements	Stakeholder involvement in the trade studies	Enable modeling and Simulation for Stakeholders to experience the interactions of solution elements and the environment	Utilizing multiple techniques in conjunction such as Software, SE and Stakeholders to define the boundaries
Complexity in Communication between various components	Requirements related to faults of the system under various scenarios to be captured	a. Minimal communication between the elements of complex systems b. Model with system dynamics	Various modelling techniques to assess the ability of solution against assumptions	Interdependency database between the Systems that is applicable for all major design decisions

Table 3: SE Approaches to provide solutions to design CS

Complexity / Approaches	Requirements Elicitation and Development	Trade Studies	Architecture Design	Development Method
Complexity in designing CS	Use multiple models at different scales simultaneously to describe a system including investigation and computational modeling to achieve the following goals: • Create intuition into the actions of specified requirements • Provide better trade studies and to advise trade-off conclusions	Different models to confirm appropriate results of trades that are discovered at different levels of buildup	Highlight selection of strong and Provide elements which is having ability to change to different conditions and structures to meet current requirements	System of Systems Engineering approaches help to coordinate components of systems

Complexity / Approaches	Requirements Elicitation and Development	Trade Studies	Architecture Design	Development Method
Growing Behaviors in Complex Solution System	Make the most of description of prominent properties in different circumstances and assignment definition. Gathering requirements from different stakeholders perspectives and at different levels of accumulation Confirm requirements and constraints at all levels to understand the real value of probability	Application of different models to confirm appropriate results of trades are discovered at different levels of accumulation	Establish a feedback mechanisms to improve ability of the system and system elements to adjust to any kind of environment Recognize the restrictions to the decomposition-based methods	Carry out development events always within context of the whole system Use collective development processes so that data about design pronouncements are evident throughout the project Prototyping and complete testing are critical to reconnoiter and check for the appearance of developing behavior
CS Deployment	Use soft systems techniques to cover the features of the organized solution and its internal structure and information movements Use problem definition methods from an evolutionary System of Systems Engineering	Trade conditions required to check ease of training, value cost and logistical support over procurement cost Model system with inherent algorithms	Whenever it is possible, use system to spontaneously arrange its components and self-healing elements	Use soft systems techniques to cover the issues, participate stakeholders, recognize methods to advance the installed system and to accomplish stakeholder agreement to the solution System of Systems Engineering to identify effectiveness and cost of using and adjusting inheritance systems
CS Evolution	Emphasis on competences, not requirements	In trade studies, use principles toughness and the ability to withstand adverse conditions	Confirm that the most significant system elements are capable of being re-configured easily with other elements in the future to satisfy operational need to have clear and available interfaces	Provide special importance to understand the problem at each stage of the iterative process, Alter to add an incremental capability and observe cautiously over time to see whether there is development. If not, different approaches to be tried

6. CONCLUSION

This survey paper explains SE process, RAM engineering and MBSE methodologies. Different MBSD designs focus on specific areas, and various designers from different disciplines are provided as a gist. Moreover, failure of some products originated in undiscovered couplings of functions in the design stage. But MBSE is lacking to provide proper models for non-functional requirements and some stakeholders find it challenging to understand the model based approach. However MBSE at the same time equips SE to design systems with complexity considering various dimesions of system architecture which forms a strong base for product implementation. The complexity and approaches dimesion is provided for problem, solution and environment contexts. The future work for the authors include, applying one of the RAM techniques and deployment of the same in a live environmet for an ADAS based safety critical system, analyze the outcome with respect to state of art methods and perform a retrospective analysis.

REFERENCE

[1] Haoqi Wang Hao Li Chengtong Tang Xu Zhang Xiaoyu Wen, "Unified design approach for systems engineering by integrating model-based systems design with axiomatic design," *Wiley Online Library*, 2019

[2] Juntao Zhang, Cecilia Haskins, Yiliu Liu, Mary Ann Lundteigen, "A systems engineering–based approach for framing reliability, availability, and maintainability: A case study for subsea design," *Wiley Online Library*, 2018

[3] Object Management Group. *OMG System Modelling Language (SysML)*, Version 1.5, 2017

[4] INCOSE, "INCOSE Systems Engineering Handbook: A Guide for System Life Cycle Processes and Activities", 4th Edition, 2015

[5] INCOSE, "A Complexity Primer for Systems Engineers", 2015

[6] ISO 26262, "Road Vehicles—Functional Safety. Geneva, Switzerland: International Organization for Standardization," 2011

[7] Peter H, Hoffmann, "SysML-based systems engineering using a model-driven development approach," *Wiley Online Library*, 2008

[8] Kartic Subr, Aditi Majumder and Sandy Irani, "Greedy Algorithm for Local Contrast Enhancement of Images," *ICIAP'05 Proceedings of the 13th international conference on Image Analysis and Processing*, pp. 171-179, 2005.