Factors Affecting Systems Engineering Complexity during Developmental Phase: Systems Practitioners, Developers, and Researchers' Perspectives - Systematic Review

[1]Mahir H. Moh. Ibrahim, [2]Abdelgadir Mahmoud, [3]Roslizar Bin Mat Ali

[1]Ph.D. Candidate, [2]Senior Lecturer, [3]Associate Professor
[1]Engineering and Technology,
[1]Razak Faculty of Technology and informatics, University Technology Malaysia, Kuala Lumpur, Malaysia

***Abstract*: The Systems Engineering design process is challenged to deliver successful complex systems in multidisciplinary and heterogeneous components. Growing human needs and evolving society bring ever greater challenges in the formation of a complex and large engineered system. System complexity is related to lots of parts and large size of the system when there is difficulty in understanding how the system works or in predicting the consequences of any change that may affect the process and systems develop itself. The leadership is performing an important role to manage a complex system. Leaders should be able to set back from immediate focus and look at the desired big picture. In practice, many factors contribute to Systems Engineering complexity in this review. This study intends to explore and analyze the complexities and the factors that contribute to the complexity of the Systems Engineering design approach. The data in this study were collected systematically from several electronic scholarly databases, including the ISI Web of Science, Scopus, Wiley Online Library. This study quantified the challenges and causes of the Systems Engineer complexity. Then, the challenges were categorized into two groups, managerial and technical causes. Ultimately, seven Systems Engineering complexity factors were identified, and their impact on the Systems Engineering processes was ranked using the Pareto principle. Among the factors, rapidly emerging technology was the most significant factor contributing to Systems Engineering complexity.**

***Index Terms*: Systems Engineering, complexity factors, rapidly emerging technology, design process, development phase insert.**

I. INTRODUCTION

The need to satisfy human development and grow the prosperity of society create systems requirements at a fast pace. Responding to those requirements may lead to large and complex engineered systems, but at the same time, it should add value to society. Complexity often occurs from a considerable number of components and their connectivity [1]. Further, the system's complexity is the result of coupling interconnectivity and interdependence characteristics that address stakeholder desires. Both these properties will guide the future design of systems regardless of application domains (INCOSE, 2014). Later, Cloutier [2] introduced two more features that could influence the future of the systems design: simplicity and adaptability. However, the last two characteristics are critiqued because they do not generate a reasonable complexity for the systems, as they could indirectly be contained in both interconnectivity and independence characteristics of the system's elements. As Kossiakoff, Sweet [3], Systems Engineering is an essential design approach to realize such advanced technological complex systems.

Engineers use the Systems Engineering approach to cope with systems complexity. However, it is still a challenge for the Systems Engineer to deliver complex systems that meet the requirements and the current trends in technology growth. There are a few Systems Engineering complexity factors that are the main challenges to deliver successful systems. The factors that have high impact on the Systems Engineering design approach were identified, analyzed, and discussed in this study.

Studies over the past five years provided important literature on the challenges encountered in Systems Engineering. Pennock and Wade [4] defined ten assumptions and illusions of Systems Engineering. Their study also highlighted that traditional Systems Engineering practices depended on many of these assumptions, rather than on sound scientific knowledge. Then, Madni and Sievers [5] introduced concurrent Systems Engineering, that is, the Model-Based Systems Engineering (MBSE) that overcomes the defined ten assumptions and illusions of the Pennock and Wade [4].

This review used exploratory and interpretive methodology to investigate the following research objectives:

1. Identify the factors that contribute to Systems Engineering process complexity during the developmental phase in Systems Engineering life-cycle.
2. Analyze the factors that contribute to the complexity of the design during the developmental phase in the Systems Engineering life-cycle.

II. METHODOLOGY

This study aims to explore the phenomena of the complexity in the Systems Engineering design approach. Accordingly, the exploratory sequential design by Creswell and Guetterman (2018) was used throughout the research objectives. Firstly, the data was qualitatively collected through a systematic review of the studies that related to the problem. The factors of the system engineering complexities were then identified for analysis. The researcher has combined and adopt procedures for planning and conducting the systematic literature review from the concept of Xiao and Watson [6] and the perspective of Carrión, González [7]. As a result, appropriate, practical, and useful procedures have been introduced throughout the study in the following sections.

DATA COLLECTION

The primary data in this study were mainly collected from several electronic scholarly databases, including the ISI Web of Science, Scopus, Wiley Online Library, IEEE, and International Council on Systems Engineering (INCOSE The author confined research to articles five years old or newer as of the beginning of 2020. Articles had to be published in indexed journals and written in the English language. Keywords like Systems Engineering, Challenges, Complexity, and Factors were used for preliminary search in the databases for journals focused on Systems Engineering practitioners and developers. Fig. 1 represents the structural diagram of terms variation and synonyms, followed by inclusion and exclusion criteria for refining.

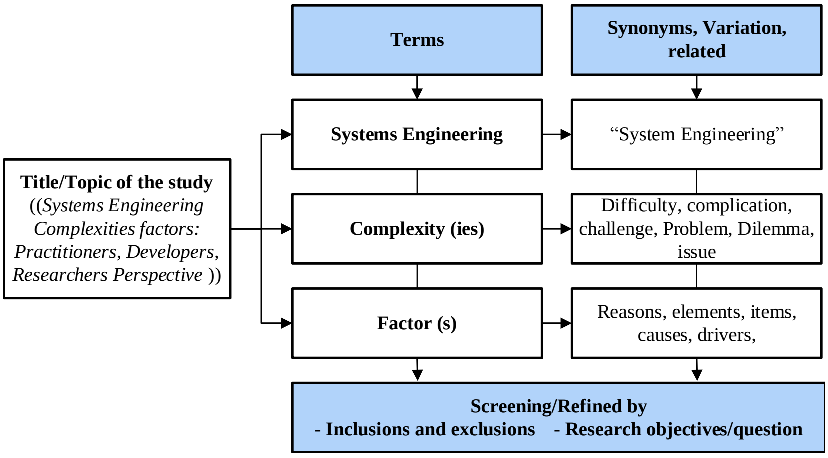

Figure 1 Structure diagram terms variation of searching

Table 1 represents the initial search script and results of the literature review studies in the field of Systems Engineering Complexities in each database.

Table 1 Search script used for each data base

Data Base source	Syntax	Results/Valid
Web of Science (WOS)	(TI=("System* Engineering") AND TI=(complex* OR difficult* OR complic* OR challenge* OR problem* OR dilemma* OR opportunit* OR issue*)) AND LANGUAGE: (English) AND DOCUMENT TYPES: (Article OR Editorial Material)	Results = 90 = 23 after removing (book chapter magazine + conferences+ review) or include only Articles and Editorials = 22 include English Languages only
Scopus	TITLE ("system* engineering") AND TITLE (compex* OR difficult* OR compli* OR challenge* OR problem* OR dilemma* OR oppprtunit*) AND PUBYEAR > 2014 AND (LIMIT-TO (DOCTYPE , "ar"))	SCOPUS Results = 55 = 15 after excluding (book chapter magazine + conferences+ review) = 15 include English Languages only
IEEE	("Document Title":"system* engineering") AND ("Document Title":complex* OR difficult* OR compli* OR challenge* OR problem OR dilemma OR opportunit*) + issue*	IEEE Results = 194 = 28 after removing (book chapter magazine + conferences+ review) = 28 include English Languages only
Wiley Online Library	"system*+engineering" in Title and "complex* OR difficult* OR compli* OR challenge* OR problem* OR dilemma* OR opportunit*" in Title	Wiley Results =198 = 139 after removing (book chapter magazine + conferences+ review = 139 include English Languages only

SCREENING – INCLUSION AND EXCLUSION CRITERIA

The population includes published articles on the Systems Engineering approach, which could be technology-based or management-based, including related journal articles, technical publications, standards, and special papers.

The Endnote references management tool has been used to manage all compiled lists of references. Fig. 2 demonstrates the flow diagram of identification, screening, eligibility, and included studies. 362 documents from the complied list at identification level were filtered to exclude theses, conference papers, book chapters, and magazines. 25 references were excluded due to duplications. Finally, one reference was removed because it was not in English. About 31% of results (178 titles and abstracts) reached the eligibility level. A systematic review quality was made to exclude non-relevant topics.

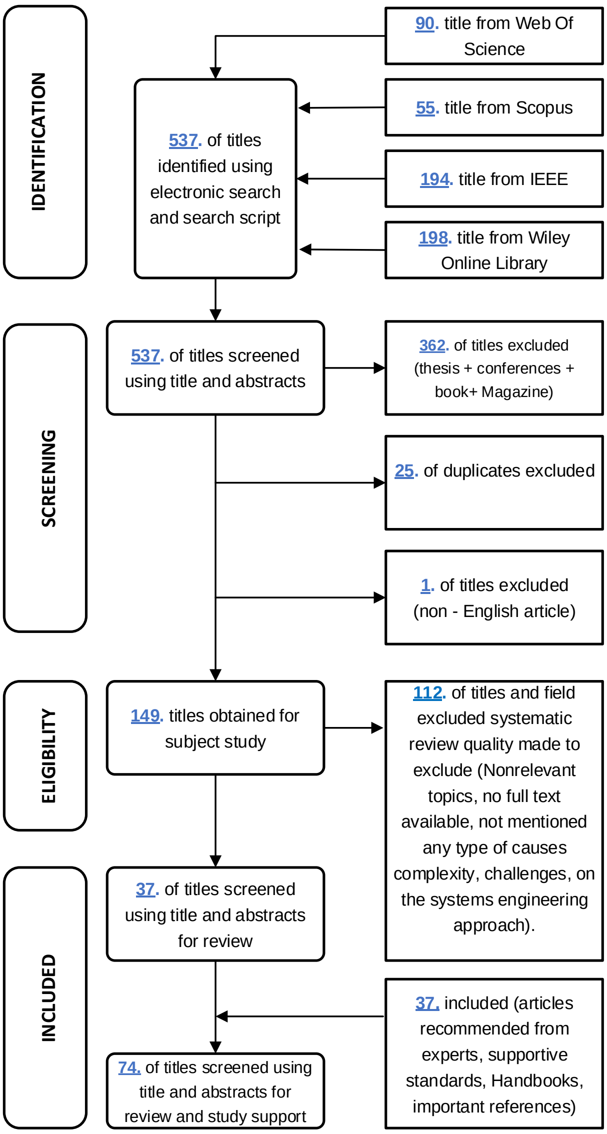

Figure 2 Flow diagram of the literature reviewed

III. COMPLEXITY AND SYSTEMS ENGINEERING

This section highlights the definition of the term complexity and determines the factors that influence the system engineering process based on the data extracted from the systematic review. Fig**.** 3 shows the flowcharts used to achieve the research objectives.

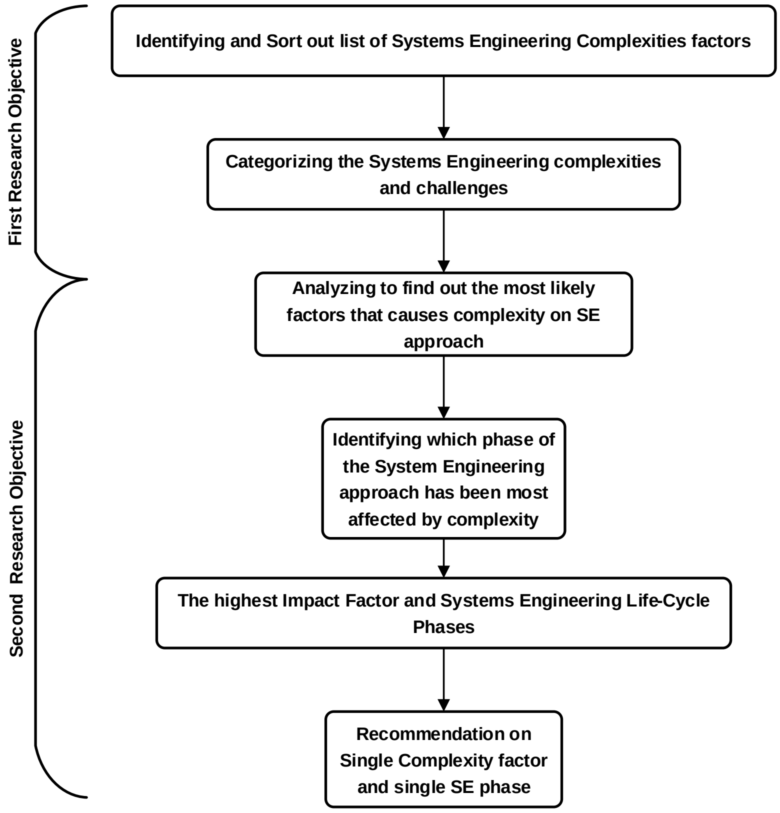

Figure 3 Flowchart of the study steps to achieve the research objectives

COMPLEXITY

Wildly divergent definitions of "complexity" have emerged; it has no singular meaning [8, 9]. Applying such a broad topic as complexity to the equally common topic of Systems Engineering is extremely difficult [10]. However, this study introduces an operational definition of complexity1,2 regarding Systems Engineering and its life-cycle. Hall [11] stated that Systems Engineering originated to deal with complexity. Nevertheless, new and emerging complications have arisen due to fast change in requirement.

Snowden and Boone [12] showed that the system could be Simple, Complication, Complex, and Chaotic. The complex systems are always taking the characteristic of Unknown Unknows in advance, such as in Systems Engineering process in Systems Engineering process a major change in the system requirement, unpredictable emerging technology used in the main system, a shift in management. At the complex system, the Cause-and-Effect relationships are so intertwined they are only evident in the late stages [12]. This feedback brings no right answer, with many competing ideas. The domain of complex systems needs creative, innovative methods [12]. Besides, leadership is performing an essential role in managing a complex system by creating environments that allow patterns to emerge and increasing levels of interaction and communication. Also, leaders should be able to set back from immediate focus and look at the desired big picture and applying holism principle [13].

[1.] Sheard and Mostashari [8] stated "*Complexity is associated with difficulty of understanding, difficulty of teasing apart the problem (or system) without destroying the emergent functionality, and difficulty of prediction and control. Complexity is also associated with large size, lots of parts, things that are densely interconnected, things that have many different types of parts.*"

[2.] Cloutier [2] defined complexity is " *a measure of how difficult it is to understand how a system will behave or to predict the consequences of changing it. It occurs when there is no simple relationship between what an individual element does and what the system as a whole will do, and when the system includes some element of adaptation or problem solving to achieve its goals in different situations.*"

White [14] introduced well processes of leadership on complex adaptive systems, while Snowden and Boone [12] said the leaders have to act differently from the previous solution during managing complex systems due to deferent variables and want to apply complexity science. Also, he introduced the leadership's framework for decision making, which could be easier to follow in the complex adaptive systems.

However, Systems Engineering principles and theories could help to deal with big picture view which is required to cater with complex systems, such as, Principle of Holism "*A system has holistic properties not manifested by any of its parts. The parts have properties not manifested by the system as a whole*"[15], and also Hitchins [16] stated, "*systems engineering addresses the whole problem, and creates the whole solution*."

SYSTEMS ENGINEERING COMPLEXITIES AND CHALLENGES – REVIEW OF EXISTING PERSPECTIVES OF PRACTITIONERS AND DEVELOPERS.

Based on the reviewed literature, the terms 'complexities,' 'challenges,' and other synonyms have the same sense. The researcher has developed descriptions and typologies to discuss the complexities and challenges of Systems Engineering. Systems Engineering challenges were appropriately categorized as external and internal challenges (Table 2). External challenges include changing global policies, regulations, or even technical patterns. Internal challenges are difficulties within the Systems Engineering processes such as bad requirement definition, lack of training, or holistic view deficiency in managing the SE process. These external and internal challenges were further classified into technical-based and management-based challenges for more specificity. The technical-based typology was used by Young, Farr [17] in Systems Engineering integration. Similarly, Sheard [18] used the same typology to further study the classification of Systems Engineering challenges. Management-based challenges were cited by Young Young, Farr [17] and INCOSE [19] as social-political complexity challenges.

Category	Type challenges	Existing challenges/Complexities	Cited authors
External	**Technology based**. Also this typology used by Young, Farr [17] as shown on (system integration base). Similarly, was used by Sheard [18].	• Rapid evolving new technologies, and big data. • Greater utilization of commercial off-the-shelf.	• D'Souza, Kossmann [20] • Roberts, Mazzuchi [21] • Farnell, Saddington [22]. • Ben Levitt MITSDM Streamed live on Apr 10, (2018). • Blanchard and Blyler [23]. • Hirshorn, Voss [24]. • Cloutier [2].
	Management based. Similar to Social-Political complexity of Young, Farr [17] and Socio-Economic of INCOSE [19].	• International competition (organizations, suppliers, subcontractor). • Resources globalization (efficient resource utilization). • Political and economic interdependence. • sharing the knowledge and technology (security issue). • Progressive Human and society needs.	• Crossley, Luan [25] • Clark [26]. • Blanchard and Blyler [23]. • University [27]. • Blanchard and Blyler [23]. • INCOSE [19]. • Young, Farr [17]. • Cloutier [2]
Internal	**Technology based**. Also this typology similar to Young, Farr [17] as shown on (system integration base). Similarly, was used by Sheard [18].	• Integration of COTS in development and production phases • Integration of Rapid Emergent Technology Items (RETIs) • Difficulty of Prediction the behavior of the system [25, 28] • Leadership	• Hoehne [29] • Crossley, Luan [25] • Curran, Allaire [28] • Farnell, Saddington [30] • Hoehne [29] • University [27] • Cloutier [2] • White [14]
	Management based. Similar to Social-Technical complexity of Young, Farr [17] and Socio-Economic of INCOSE [19].	• Poor coupling between technical and programmatic sides. • Bad holistic view. Need emphasis on all life cycle systems (whole view) • Bad definition of the system requirements. • Higher overall life-cycle costs. • last-minute changes in design • The tightness of time schedule/systems life cycle • Failure to recognize and deal with risk (example of Challenger and Columbia accident - independent technical authority)	• Hoehne [29] • Farnell, Saddington [22] • Clark [26]. • Blanchard and Blyler [23]. • Young, Farr [17]. • University [27] • Cloutier [2]

Table 2 Existing challenges for Systems Engineering

ANALYSIS OF CHALLENGES AND COMPLEXITIES RELATIONSHIPS

Fig. 4 illustrates the seven most-cited complexity factors that have direct effects on the system's life-cycle. The primary features of the information in Fig. 4 provide the context of problem background. It further shows how the problem evolved and how it reached the current status of complexities in the Systems Engineering approach.

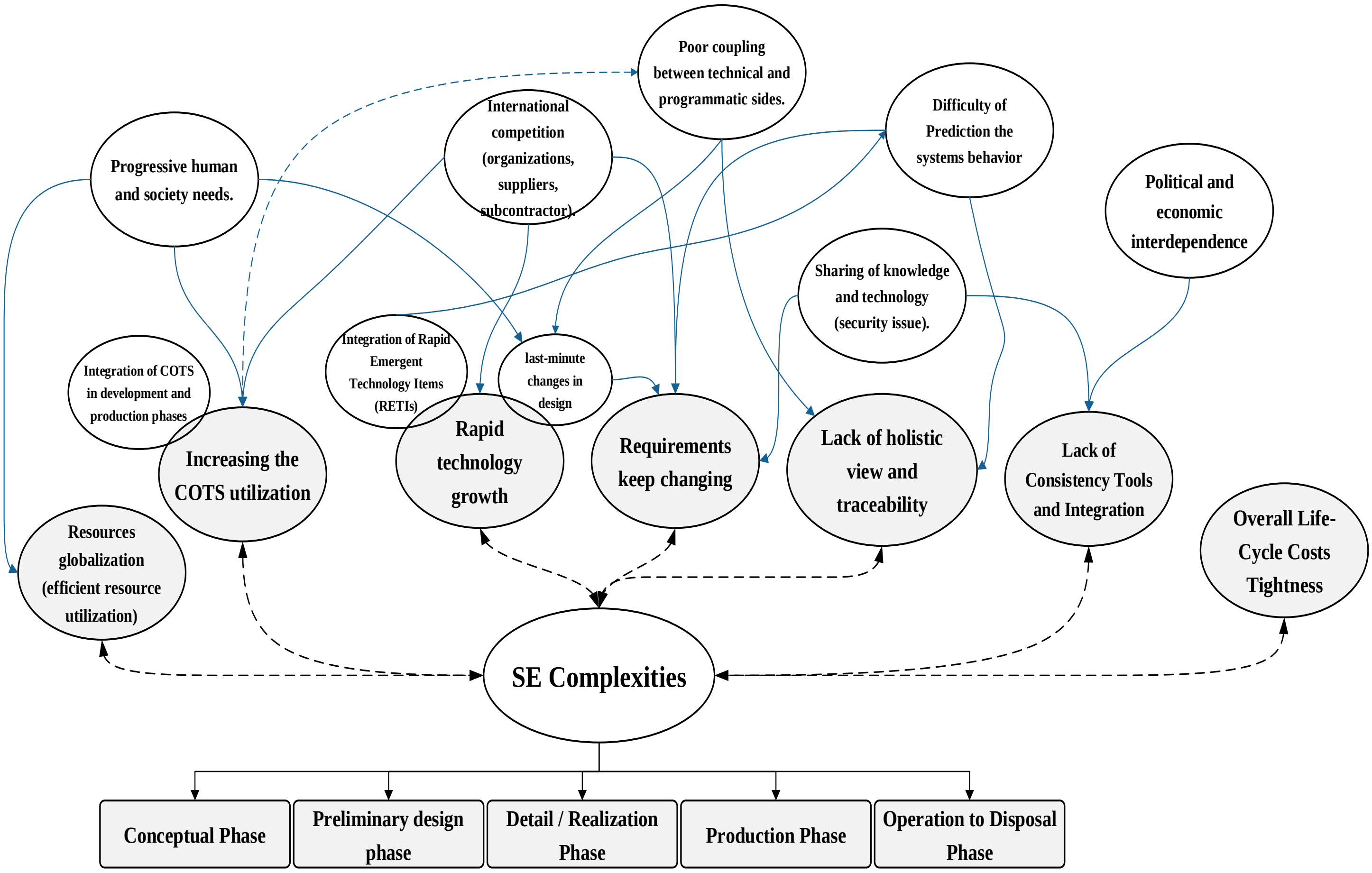

Figure 4 Relationship between existing challenges and Systems Engineering complexities

The seven complexities that have a stronger correlation with the Systems Engineering phases include:

1. Resource globalization. It is affected by human needs and last-minute change requests by stakeholders. Human always seek comfort with emerging new designs if they think that makes their lives better; nevertheless, that puts pressure on system engineers to meet these requirements while designing a system. The last-minute change is usually the consequence of a customer request or technical feasibility. Furthermore, it has facilitated the use of faster and more efficient transportation /communication means for expediting procurement and related processes [23, 31, 32].
2. Increasing the utilization of Commercial Off The-Shelf (COTS). It may be derived from the competition of the items' suppliers and the growing needs of humans and society. In addition, the international competition of parts suppliers creates a variety of options from products. If COTS is built in the system, the complications arising from integration besides alignment of specifications [23, 33].
3. Rapidly Emerging Technology. This is influenced by COTS and progressive human needs. Correspondingly, it causes difficulties in the prediction of systems performance and in integration while developing a system [3, 34-37]
4. The change in requirements. This factor influences more than four motivates, as shown in Fig. 4, all are related to change requests. In the industrial sector, inconsistent requirements and different performance objectives usually make the decision-making process more complex [38]. New requirements are a challenge to delivering efficient system [39].
5. Lack of holistic view and requirement traceability. The main concern stems from a poor coupling between the technical and management sides of the systems builders, due to a large number of subsystems and change [2, 40, 41].
6. Lack of consistency tool and integration. The inconsistency originates from the variation of a multidiscipline system of design and a large number of subsystems within traditional Systems Engineering (document-centric). Complex

systems are called interdependence systems if there is no shared management between their components/subsystems, and if they are still developing. This often results in emerging behaviour [42]. However, Systems Engineering is challenged to deliver the lowest possible interdependence in the system [2].

7. The tightness of budget for the overall system life-cycle. Basically, considerable attention has been paid to reducing the costs associated with the acquisition and procurement of systems, and little attention has been paid to the costs of system operation and support. When designing systems, it is important to observe all decisions in the context of total costs [23].

The literature listed in Table 3 provides relevant observational evidence for Systems Engineering complexity. It also shows the seven complexity factors most related to Systems Engineering design process, corresponding to the individual author's opinion and findings on each factor. More than half of the cited authors stated that the fast pace of technology growth creates complexities in the integration process of Systems Engineering. Furthermore, three factors- lack of consistency tools and integration, resources globalization, and lack of holistic view and traceability of the requirements- were cited more than ten times by different authors. Two factors- bad requirements definition and overall life-cycle- have shown the lowest impact on the Systems Engineering process as they were stated less than ten times. Nevertheless, several works of literature have dealt with the concept requirements definition criteria as an essential question on Systems Engineering in other areas out of this study's scope.

Table 3 Factors that contribute to the Systems Engineering complexity by different authors

No.	Authors	**Effect of Systems Complexity Faced by Systems Practitioner, Developer, Researchers**						
		Rapidly Emerging Technology (21)	Globalization Market Place (13)	COTS Using (10)	Lack of Holistic View and Traceability On SE (12)	Bad Definition of the Requirement (07)	Overall Life-Cycle Costs Tightness (02)	Lack of Consistency Tools and Integration (16)
1	Curran, Allaire [28]	■	■			■		■
2	Winzer [31]		■					
3	Sheard [43]			■				
4	MacKenzie, Bryden [44]			■				
5	Wu, Gouyon [33]			■				
6	Roberts, Mazzuchi [21]	■	■					
7	Shortell [34]	■	■					
8	Hoehne [29]				■			■
9	Crossley, Luan [25]							■
10	University Lecture Notes [27].	■	■	■	■			■
11	Madni and Sievers [5].	■			■	■		■
		Technology Growth (21)	Globalization Market Place (13)	COTS Using (10)	Lack of Holistic View and Traceability On SE (12)	Bad Definition of the Requirement (07)	Overall Life-Cycle Costs Tightness (02)	Lack of Consistency Tools and Integration (16)
12	Farnell, Saddington [22].	■			■			
13	D'Souza, Kossmann [20]	■	■			■		
14	NDIA [45].	■		■	■			■
15	Blanchard and Blyler [23].	■	■	■	■		■	■
16	Eisner [46].	■	-	■	■			■
17	Haskins, Forsberg [47].	■			■	■		■
18	INCOSE [19].	■	■					
19	Ashby, Blessner [48].	■	■				■	

No.	Authors	**Effect of Systems Complexity Faced by Systems Practitioner, Developer, Researchers**						
20	Sheard [18].		■					
21	Hirshorn, Voss [24].	■		■				
22	Reichwein and Paredis [49].			■				
23	Horváth, Tepjit [50].				■			
24	Ben Levitt MITSDM Streamed live on Apr 10, (2018).	■				■		
25	Blackburn, Verma [51].		■					■
26	Cloutier [2].	■	■	■	■	■		■
27	Ward, Rossi [52].	■						
28	Sales and Becker [53]							■
29	Lankhorst [32].	■	■					
30	Mordecai, Dori [54].					■		
31	Liebel, Marko [55].							■
32	Zdravković and Panetto [56].	■						
33	Walworth, Yearworth [57].				■			
34	Wheaton and Madni [58].							■
35	Pennock and Wade [4].				■			■
36	George Mathew, Liscouet-Hanke [59].	■						■
37	Kenett, Zonnenshain [60]	■						

The Pareto principle, or more accurately, the rule of "80/20," which explains cause and effect, was used in this study. It is a statistical analysis tool used to select a limited number (20%) of overall variables for decision-making to achieve a meaningful overall effect (80%) [61, 62].

The Pareto principle was used to determine the factor that has the highest impact on Systems Engineering life-cycle during the developmental phase. The frequency and accumulative percentage of the factors that have an impact on the SE complexity are shown in Fig. 5.

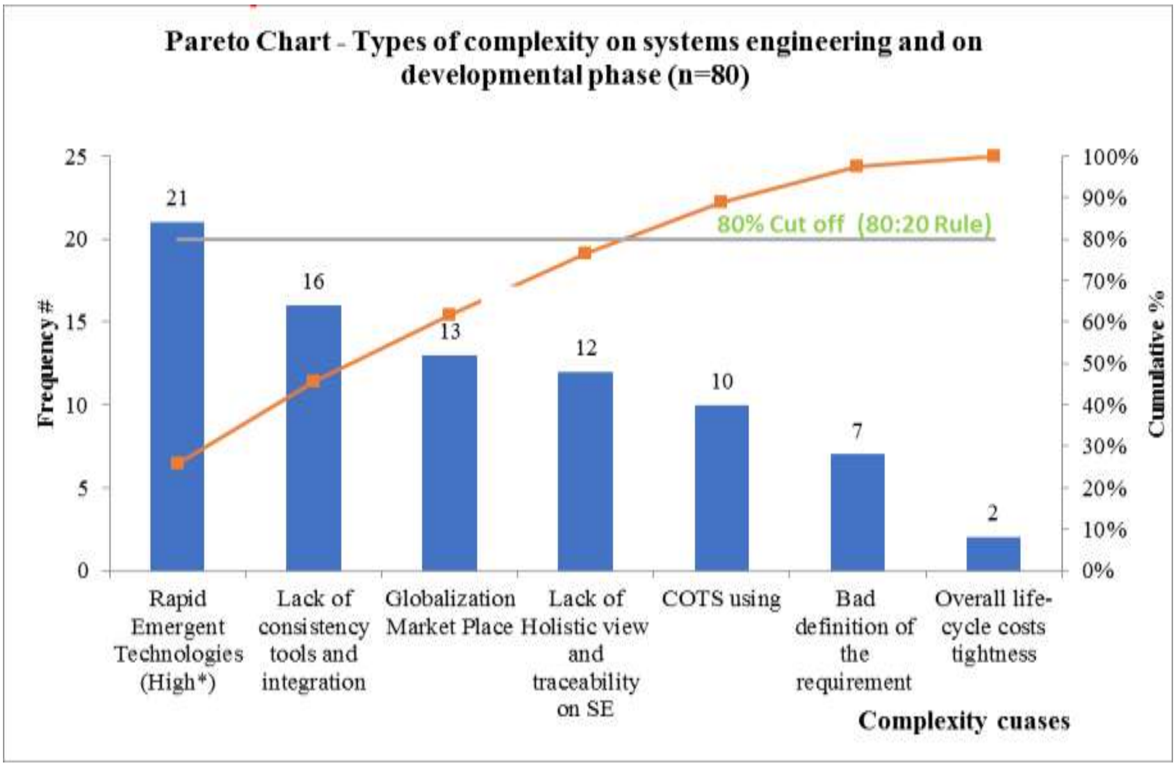

Figure 5 Pareto Chart – Factors contributing to System Engineering Complexity (N=80)

The results showed that technology growth (*Rapidly Emerging Technology Items*) was the most significant factor (Fig. 5). Moreover, the Rapidly Emerging Technology Items lay above the line of the percentage of citation accumulations, which indicates that if a researcher reduces the impact of this factor, they are likely to solve 80 percent of Systems Engineering complexities problems.

Table 4 shows the Systems Engineering phases beside short descriptions of their tasks and levels of the system. It also reflects the associated complexity factors for the design phases of Systems Engineering that were obtained from the preliminary analysis of Fig. 5.

Table 4 Systems Engineering phases and challenges throughout life cycle

	Conceptual design	**Preliminary design**	**Detail design and development**	**Production/ construction**	**Operational use and system support**
Activities description [24, 63, 64]	Need identification; functional constraints; R&D needs analysis; maintenance and support concept; selection of technological approach; evaluation of feasible technology applications; system functional definition; system/program planning.	Functional analysis; trade-off studies; allocation of requirements; synthesis; preliminary design; testing and evaluation of design concepts (early prototyping); contracting; acquisition plans; implementation of programs; major suppliers.	Subsystem/component design; trade-off studies and of alternatives evaluation; development of engineering and prototype models; verification of manufacturing and manufacturing processes; developmental testing and evaluation; supplier activities; production planning.	Production or construction of system components; production activities of suppliers; acceptance testing; distribution and operation of systems; testing and evaluation development / operational; system assessment.	System operation in the user environment; maintenance and logistic support; operational testing; system improvement modifications; contractor support; system assessment (field data collection and analysis).
Level of the system	**Primary System Level**	**Subsystem Level**	**Detailed level/Components level**	**Items/Parts**	
	Example: aircraft and/or related systems.	Major aircraft subsystems, including hydraulic, electrical, avionic, power plant, fuel, air conditioning, structure, seats.	Components that include wing, fuselage, tail, landing.	Parts that include fittings, fasteners, blades, propellers, screws, nuts, ribs, spars, frames, stiffeners, skin, shafts, wires.	Feedback and modification for improvement or optimization
Associated complexities		**Overall lifecycle tightness of cost** **COTS utilization** **Globalization market place and competitions**	**Bad definition of the requirement** **lack of holistic and consistency** **Lack of Consistency Tools and Integration** **Rapidly Emerging Technology Items**		Complexities Density

All the factors contributing to the SE complexities extend to more than one stage. Three factors- "the overall cost of the life-cycle," "lack of a holistic view," and "poor definition of requirements-" influence all phases of the System Engineering design approach. Rapidly emerging technologies and wrong definition of requirements weighted highest among all other factors.

Finally, any changes to the requirements, items concepts, or technologies upgrade will lead to a redesign of some subsystems, which in turn, will increase the cost of iterations periods.

The factors contributing to the SE complexity, agreed on by the authors, practitioners, and designers, will be discussed in the following sections.

1. Rapidly Emerging Technology

There is no single definition of the central concept of Emerging Technology [65]. In this study, *Rapidly Emerging Technology* refers to the advancement of the items or components used during the development phase of the systems. The rapid technological advancement discussed by Blanchard and Blyler [23] shows that technology is growing faster than our ability to control and manage it through a traditional Systems Engineering approach. Systems designers should consider existing and potential future improvements in technologies to the system while designing it. Otherwise, there is a significant possibility that the main system will be out of trend.

The International Council On Systems Engineering INCOSE [19]) has named the following technologies as ones that will create challenges to Systems Engineering:

- sensor technologies:
- material science
- miniaturization
- human-computer interaction technologies
- computational power

Additionally, it is very useful to consider electrification and hybridization technologies as a fast-growing technologies. This adds a level of challenges to current systems under development and the Systems Engineering design approach as these technologies are under development and making very real progress (Brelje & Martins, 2019; Schäfer et al., 2019).

One of the problems of ongoing technologies is technology insertion in the man system. The insertion scenario of new technologies while designing using the Systems Engineering approach has not been mentioned [66]. Moreover, some of the technology insertion processes are still struggling and may cause significant reworking: for example, the lithium-ion batteries in the Boeing 787 Dreamliner model [67].

2. Lack of Consistent Tools

Madni and Sievers [5] indicated that traditional Systems Engineering models such as Waterfall, V-model, and Incremental were facing inconsistency in addressing the heterogeneity of subsystems. As the systems continued to scale and increase in application complexity, they were unable to maintain consistency and assure traceability during systems development at the same level.

Based on the data that was extracted from the literature listed in Table 3, the causes of inconsistency in systems integration could be summarized as follow:

- a large number of systems levels,
- multidiscipline in systems,
- sequential of traditional Systems Engineering. Even though concurrent Systems Engineering has overcome such deficiencies, it still lacks full functionality, and

A study by Madni and Sievers [5] presented Model-Based Systems Engineering (MBSE) as a methodology equipped with suitable tools like repository (single source of truth) for data and information exchange between systems team builders. This model may overcome most of the traditional Systems Engineering deficiencies.

3. Globalization and marketplace competitions

Globalization means there are interdependencies in the world [23]. In depth, many systems are currently being developed by several departments at different places, by multiple suppliers, and through multiple organizations [68]. It is facilitated by rapid and improved communications practices [18]. Globalization affects the Systems Engineering phases that involve multisource components and items. Due to the increasing global competition caused by globalization, systems developers have better access to global resources to fulfil systems requirements. A strong correlation has been observed between globalization and the utilization of Commercial Off The-Shelf (COTS), which further improves the accessibility of the required resources in Systems Engineering [31]. However, multiple studies identified globalization as the complexity factor to multi-resource management in the Systems Engineering life cycle [27]. In the development phase of the Systems Engineering life cycle, the designer needs to perform preliminary prototyping and contracting according to major Systems Engineering standard frameworks from the Department of Defence (DoD) or the National Aeronautics and Space Administration (NASA) [24]. Otherwise, the traditional Systems Engineering approach might experience downstream failure on verification and validation, or even in the production phase.

4. Lack of holistic view

Systems Engineering is defined as a holistic and integrative discipline (Hirshorn, Voss [24]. Nevertheless, Cloutier [2] from the INCOSE global community noted that one of the factors that made Systems Engineering more challenging was the lack of a holistic view by practitioners and system developers. The development phase is one of the important phases of the systems life-cycle, where most of the engineering developments happen. A holistic view is an essential personal skill for practitioners and systems developers in planning, organizing, and realizing the system-of-interest. This will remain a challenge if enterprises wish to be competitive in the current technology trend.

A holistic approach helps to reduce risks and difficulties in managing tasks in the development phase and improve the communication between the systems levels. Furthermore, Cloutier [2] stated that use of Holism principle is essential in Systems Engineering to reduce individualization risk. Currently, the Systems Praxis Framework, which was developed by INCOSE and International Society for the Systems Sciences (ISSS), is the potential solution for Lack of holistic to today's complex systems [2].

Possible explanations of the consequences of the "lack of holistic view" in Systems Engineering is unable to explain the behavior of the overall systems by individual parts [50].

5. Commercial Off The Shelf (COTS) utilization

Increasing numbers of systems have adopted COTS to lower the initial procurement costs and shorten the acquisition cycle (Blanchard and Blyler, 2016; Eisner, 2008). This is because COTS usually makes use of the latest commercial technology, which will be replaced by new technology innovation in a short period of time, and are taken as consumables [69].

To its advantage, COTS could be selected and implemented for technical (shortening the developmental phase), organizational (reducing the overall cost of the developmental phase), or strategic reasons (access to technology not available internally) [33]. Recently, professional COTS often comes with supporting documentation such as proof of verification, validation conformance, and manufacture specifications or fact sheet. It is always recommended to review the specifications to ensure the COTS fit the requester's requirements [24].

On the other hand, the advantages of COTS challenged by integration concerns such as performance (what is supposed to do), compatibility (no standards), product assessment (uncertainty of meeting the required needs), or supplier behavior (agreements promises false) [33]. Additionally, MacKenzie, Bryden [44] observed that many companies struggle with COTS integration into the Systems Engineering processes, the possible reason the employee is unfamiliar with that COTS.

Moreover, stakeholders occasionally order COTS, which indicates that they have decided to apply a readily available solution to their system without first validating that solution [54].

Furthermore, risks associated with the use of COTS during system life-cycle include the obsolescence of models and of improvements in system interfaces. The competition of COTS providers makes individual components obsolete within two years [34]. The incompatibility between two or more systems requirements in COTS specifications may cause further complications to systems engineers. It is a nontrivial task to tackle this type of complexity, as it depends on the experience and design expertise of hardware and software engineers [46].

According to the literature listed in Table 3, COTS usage was an unavoidable situation in many cases. Multisource of COTS generates complexity to integration in Systems Engineering processes during the development phase due to deferent standards used. It will be difficult to integrate new COTS during the development process, as this may lead to the re-integration and re-assessment and re-validation of the system under construction.

6. Bad Definition of the Stakeholder Requirement

Mordecai, Dori [54] defined the Stakeholder requirements as ideas, expectations, requests, a set of needs, goals, assumptions, guidelines, preferences, objectives, constraints, intentions, or desires. Requirements are the needs or demands of the stakeholder collected as statements to constrain and identify a system or process. The ideal requirements are clear, unambiguous, consistent, unique, traceable, verifiable, and "SMART" (Specific, Measurable, Attainable, Realizable, Time-bounded) [70]. However, the inadequate definition of the requirement for stakeholders indicates that the requirement is improperly collected or elicited, which leads to developmental phase difficulties, extra cost for a delayed change request, functional analysis, or issues in integration and systems evaluation.

Practically, requirements are the input to the design process at the beginning of systems formulation, while specifications are the output of the development phase. In other words, the development phase is begun with well-defined requirements [23, 24]. Yet, some inexperienced developers are unable to differentiate a requirement from a specification.

Continuous monitoring of the requirements during the Systems Engineering life-cycle results in internal complexity as it requires controlled traceability, and hence adds more tasks to systems developers. de Weck [71] and Hirshorn, Voss [24] proposed some tools that could assist the requirement monitoring process, including Excel spreadsheets, professional commercial tools like DOORS for large complex systems, and metadata.

While traditional sequential Systems Engineering is unable to address the complexity from continuous requirements monitoring that involves rapid and dynamic changes of input, a holistic view and MBSE can reasonably overcome this issue [5].

Standard definitions to reduce the risk of bad Stakeholder requirement definition are as follows:

- ISO / IEC 15288 (IEEE STD 15288 – 2008)
- System Engineering Handbook, Version 3.1 Working Group requirements (http:/www.incose.org/ChaptersGroups/WorkingGroups/processes)
- System life cycle process (6.4.1 Stakeholder Requirements Definition Process)
- NASA Systems Engineering Handbook

7. Overall Life-Cycle Cost – Tight Budget

Briefly, a Systems Engineering life-cycle cost is the expenses from all the systems phases [2].

Usually, a tight project budget has a positive impact on the project/system owners, the sponsor, and other stakeholders. In this case, however, systems engineers and developers are challenged to deliver the system within that limited cost range. As a result, the cost tightness combined with a poor definition of the requirements and a dynamically changing environment makes task management complicated.

When calculating the overall life-cycle cost, the systems engineer needs to understand that any of the above-mentioned processes might need to be repeated until the desired specification is achieved. Moreover, rapid and frequent change in technology and requirements will affect the life-cycle cost analysis of the systems of interest as well, even though it may lead to better performance.

Concisely, overall life-cycle cost is causing complexity in Systems Engineering in the following ways:

- requirement of minimum life-cycle cost,
- dependency of the lifecycle cost in decision making and design reviews output, and
- dependency of life-cycle cost in making the decision that relates to the change of technology in the systems-of-interest.

IV. Conclusion

A Systems Engineering complexity is defined by different sources as a measure of the difficulty in understanding the behavior of a system or in predicting the consequences of a change. This study identified and analyzed factors that lead to the complexity of Systems Engineering throughout the phases of the life-cycle of Systems Engineering.

Complexity factors were found to be linked to each other and intertwined with the Systems Engineering stages or processes. This creative synthesis is essential to produce an integrated system that can meet the end-user requirements.

Although other factors are crucial, rapid emergent technology has the highest impact on Systems Engineering complexity, particularly during the developmental phase in complex systems like Aviation industry, because any changes accrue during design process lead to verification and revalidation and reintegration as well. MBSE can be used to resolve some of the SE complexity issues. Still, some of the factors remain as challenges to the traditional Systems Engineering.

In order to integrate cutting edge technology into the Systems Engineering processes, a system engineer or developer needs to follow the technology development closely. The rapid change in emergent technologies happens during systems development will lead to repetition in requirements verification, specifications validation, subsystems or components design, trade-off studies, alternatives evaluation, engineering and prototype models development, production planning, and tests and evaluation development. The available Systems Engineering frameworks or guidelines lack means to accommodate and address the issue of rapidly emerging technologies.

Some questions remain unanswered, especially in interconnectivity improvement. Improving system interconnectivity eliminates the erratic behavior of complex multidiscipline components and helps to produce a better systems design, especially a smart and dynamic systems solution. However, in order to improve system interconnectivity, external technical-based challenges such as globalization and COTS need to be tackled first.

MBSE has the advantage of a holistic view and traceability over traditional document-centric Systems Engineering. However, the majority of the companies do not use MBSE due to a lack of training and experience in their employees.

References

[1] Sheard, S.A. and A. Mostashari, 7.3.1 A Complexity Typology for Systems Engineering. INCOSE International Symposium, 2010. 20(1): p. 933-945.

[2] Cloutier, R.J., Guide to the Systems Engineering Body of Knowledge (SEBoK). 1.91. 2018, The Trustees of the Stevens Institute of Technolog.

[3] Kossiakoff, A., et al., Systems engineering principles and practice. Vol. 83. 2011: John Wiley & Sons.

[4] Pennock, M.J. and J.P. Wade, The top 10 illusions of systems engineering: A research agenda. Procedia Computer Science, 2015. 44: p. 147-154.

[5] Madni, A.M. and M. Sievers, Model based systems engineering: Motivation, current status, and research opportunities. Systems Engineering, 2018. 21(3): p. 172-190.

[6] Xiao, Y. and M. Watson, Guidance on Conducting a Systematic Literature Review. Journal of Planning Education and Research, 2017. 39(1): p. 93-112.

[7] Carrión, P.V.T., et al. Methodology for systematic literature review applied to engineering and education. in 2018 IEEE Global Engineering Education Conference (EDUCON). 2018. IEEE.
[8] Sheard, S.A. and A. Mostashari, 6.2.1 Complexity Types: From Science to Systems Engineering. INCOSE International Symposium, 2011. 21(1): p. 673-682.
[9] Gilbert, D. and M. Yearworth, Complexity in a Systems Engineering Organization: An Empirical Case Study. Systems Engineering, 2016. 19(5): p. 422-435.
[10] Saikou Diallo, S.M., Andreas Tolk, RESEARCH AGENDA FOR NEXT-GENERATION COMPLEX SYSTEMS ENGINEERING, in Emergent Behavior in Complex Systems Engineering. 2018. p. 370-389.
[11] Hall, A.D., A methodology for systems engineering. 1962: Van Nostrand.
[12] Snowden, D.J. and M.E. Boone, A leader's framework for decision making. Harvard business review, 2007. 85(11): p. 68.
[13] White, B.E. and B.G. McCarter, Emergence of SoS, sociocognitive aspects, in System of systems engineering—Principles and applications. 2009, Taylor & Francis Group: Boca Raton.
[14] White, B.E., On leadership in the complex adaptive systems engineering of enterprise transformation. Journal of Enterprise Transformation, 2015. 5(3): p. 192-217.
[15] Adams, K.M. and J. Mun. The application of systems thinking and systems theory to systems engineering. in Proceedings of the 26th National ASEM Conference: Organizational Transformation: Opportunities and Challenges. 2005. American Society for Engineering Management Rolla, MO.
[16] Hitchins, D., What Are the General Principles Applicable to Systems? INSIGHT, 2009. 12(4): p. 59-63.
[17] Young, L.Z., et al., The role of complexities in systems engineering cost estimating processes. 2010.
[18] Sheard, S.A. 5.2. 1 systems engineering complexity in context. in INCOSE International Symposium. 2013. Wiley Online Library.
[19] INCOSE, A., A world in motion: systems engineering vision 2025, in International Council on Systems Engineering. 2014.
[20] D'Souza, A., M. Kossmann, and S. Watts, The enabling role of Configuration Management for the Systems Engineering of tomorrow's complex Systems and Systems of Systems. INCOSE International Symposium, 2016. 26(1): p. 2180-2194.
[21] Roberts, B., T. Mazzuchi, and S. Sarkani, Engineered Resilience for Complex Systems as a Predictor for Cost Overruns. Systems Engineering, 2016. 19(2): p. 111-132.
[22] Farnell, G., A. Saddington, and L. Lacey, A new systems engineering structured assurance methodology for complex systems. Reliability Engineering & System Safety, 2019. 183: p. 298-310.
[23] Blanchard, B.S. and J.E. Blyler, System Engineering Management. 2016: John Wiley & Sons, Inc.
[24] Hirshorn, S.R., L.D. Voss, and L.K. Bromley, Nasa systems engineering handbook. 2017.
[25] Crossley, W.A., et al., Optimization problem formulation framework with application to engineering systems. Systems Engineering, 2017. 20(6): p. 512-528.
[26] Clark, J.O. System of systems engineering and family of systems engineering from a standards, V-model, and dual-V model perspective. in 2009 3rd Annual IEEE Systems Conference. 2009. IEEE.
[27] University, T.O. Systems engineering: challenging complexity. 2019 17/05/2019 [cited 2019 12/10/2019]; Available from: https://www.open.edu/openlearn/science-maths-technology/computing-ict/systems-engineering-challenging-complexity/content-section-0?active-tab=description-tab.
[28] Curran, Q.C., D. Allaire, and K.E. Willcox, Sensitivity analysis methods for mitigating uncertainty in engineering system design. Systems Engineering, 2018. 21(3): p. 191-209.
[29] Hoehne, O., The SoS-VEE Model: Mastering the Socio-Technical Aspects and Complexity of Systems of Systems Engineering (SoSE). INCOSE International Symposium, 2016. 26(1): p. 1494-1508.
[30] Farnell, G.P., A.J. Saddington, and L.J. Lacey, A new systems engineering structured assurance methodology for complex systems. Reliability Engineering & System Safety, 2019. 183: p. 298-310.
[31] Winzer, P., Generic System Description and Problem Solving in Systems Engineering. IEEE Systems Journal, 2017. 11(4): p. 2052-2061.
[32] Lankhorst, M., Enterprise architecture at work. Vol. 352. 2017, Springer, Berlin, Heidelberg: Springer.
[33] Wu, Q., et al., Use of Patterns for Know-How Reuse in a Model-Based Systems Engineering Framework. IEEE Systems Journal, 2020: p. 1-12.
[34] Shortell, T.M., INCOSE Systems Engineering Handbook: A Guide for System Life Cycle Processes and Activities. 2015: John Wiley & Sons.
[35] Sousa-Poza, A., C. Keating, and S. Kovacic, Systems engineering: evolution and challenges. International Journal of System of Systems Engineering, 2015. 5(4): p. 379-399.
[36] Saravi, S., et al., A Systems Engineering Hackathon – A Methodology Involving Multiple Stakeholders to Progress Conceptual Design of a Complex Engineered Product. IEEE Access, 2018. 6: p. 38399-38410.
[37] Wheaton, M.J. and A.M. Madni, Model-Based Approach for Resilience and Affordability Tradeoff Analysis, in AIAA Scitech 2019 Forum. 2018, INCOSE.
[38] 38. Buzuku, S., et al., A Case Study of Complex Policy Design: The Systems Engineering Approach. Complexity, 2019. 2019.
[39] de Graaf, R., et al., Value Engineering as a Specialty for Systems Engineering: Exploring Opportunities. INSIGHT, 2019. 22(1): p. 41-44.
[40] Sitton, M. and Y. Reich, Enterprise Systems Engineering for Better Operational Interoperability. Systems Engineering, 2015. 18(6): p. 625-638.

[41] Chai, H.F. and Q. Sun, Research on the Giant and Complex Financial Information System Engineering Management from the Perspective of Meta-synthesis Methodology-with the Bank Card Information Exchange System Engineering as a Sample. Frontiers of Engineering Management, 2016. 3(4): p. 404-413.
[42] Tolk, A., S. Diallo, and S. Mittal, COMPLEX SYSTEMS ENGINEERING AND THE CHALLENGE OF EMERGENCE, in Emergent Behavior in Complex Systems Engineering. 2018, John Wiley & Sons, Inc. p. 78-97.
[43] Sheard, S.A., Evolution of systems engineering scholarship from 2000 to 2015, with particular emphasis on software. Systems Engineering, 2018. 21(3): p. 152-171.
[44] MacKenzie, C.A., K.A. Bryden, and A.A. Prisacari, Integrating narratives into decision making for complex systems engineering design issues. Systems Engineering, 2020. 23(1): p. 65-81.
[45] NDIA, S.E.D., Top five systems engineering issues in defense industry, in Vers 9, 1/23/03. 2003, National Defense Industrial Association: USA.
[46] Eisner, H., Essentials of project and systems engineering management. 2008: John Wiley & Sons.
[47] Haskins, C., et al. Systems engineering handbook. in INCOSE. 2007.
[48] Ashby, D., P. Blessner, and B. Olson, An investigation into the impact of applying systems engineering life-cycle processes during new product development and the economic growth of the diversified industrial sector. Journal of Enterprise Transformation, 2019: p. 1-23.
[49] Reichwein, A. and C.J.J. Paredis. OVERVIEW OF ARCHITECTURE FRAMEWORKS AND MODELING LANGUAGES FOR MODEL-BASED SYSTEMS ENGINEERING. in Proceedings of the ASME 2011 International Design Engineering Technical Conferences & Computers and Information in Engineering Conference. 2012.
[50] Horváth, I., S. Tepjit, and Z. Rusák. Compositional engineering frameworks for development of smart cyber-physical systems: A critical survey of the current state of progression. in ASME 2018 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference. 2018. American Society of Mechanical Engineers.
[51] Blackburn, M., et al., Transforming systems engineering through model centric engineering. 2018, Stevens Institute of Technology Hoboken United States: Stevens Institute of Technology Hoboken United States.
[52] Ward, D., et al. The Metamorphosis of Systems Engineering through the evolution of today's standards. in 2018 IEEE International Systems Engineering Symposium (ISSE). 2018. IEEE.
[53] Sales, D.C. and L.B. Becker. Systematic literature review of system engineering design methods. in Brazilian Symposium on Computing System Engineering, SBESC. 2019.
[54] Mordecai, Y., D. Dori, and Ieee, Model-Based Requirements Engineering: Architecting for System Requirements with Stakeholders in Mind. 2017 Ieee International Symposium on Systems Engineering. 2017, New York: Ieee. 143-150.
[55] Liebel, G., et al., Model-based engineering in the embedded systems domain: an industrial survey on the state-of-practice. Software & Systems Modeling, 2018. 17(1): p. 91-113.
[56] Zdravković, M. and H. Panetto, The challenges of model-based systems engineering for the next generation enterprise information systems. Information Systems and e-Business Management, 2017. 15(2): p. 225-227.
[57] Walworth, T., et al., Estimating project performance through a system dynamics learning model. Systems Engineering, 2016. 19(4): p. 334-350.
[58] Wheaton, M.J. and A.M. Madni, Model-Based Approach for Resilience and Affordability Tradeoff Analysis, in AIAA Scitech 2019 Forum. 2019.
[59] George Mathew, P., S. Liscouet-Hanke, and Y. Le Masson, Model-Based Systems Engineering Methodology for Implementing Networked Aircraft Control System on Integrated Modular Avionics – Environmental Control System Case Study. 2018, SAE International.
[60] Kenett, R.S., A. Zonnenshain, and R.S. Swarz, Systems Engineering, Data Analytics, and Systems Thinking: Moving Ahead to New and More Complex Challenges. INCOSE International Symposium, 2018. 28(1): p. 1608-1625.
[61] Kiremire, A.R., The application of the pareto principle in software engineering. Consulted January, 2011. 13: p. 2016.
[62] Ng, J.J., The 80/20 Principle Thinking as a tool for Project Management. IEEE Engineering Management Review, 2019: p. 1-1.
[63] Enrico, Z., et al., Application of reliability technologies in civil aviation: Lessons learnt and perspectives. Chinese Journal of Aeronautics, 2019. 32(1): p. 143-158.
[64] Sadraey, M.H., Aircraft design: a systems engineering approach. 2013.
[65] Rotolo, D., D. Hicks, and B.R. Martin, What is an emerging technology? Research Policy, 2015. 44(10): p. 1827-1843.
[66] Saravi, S., et al., A Systems Engineering Hackathon - A Methodology Involving Multiple Stakeholders to Progress Conceptual Design of a Complex Engineered Product. Ieee Access, 2018. 6: p. 38399-38410.
[67] Oyeniya, A., Certification challenges for emerging technologies in aviation. 2018.
[68] Albarello, N. and H. Kim. Application of an MBSE Approach for the Integration of Multidisciplinary Design Processes. in Complex Systems Design & Management. 2014. Cham: Springer International Publishing.
[69] Hitchins, D.K., Systems engineering: a 21st century systems methodology. 2008: John Wiley & Sons.
[70] Szejka, A.L., et al., A Method for Formalizing Requirements Interoperation in Complex Systems Engineering. Insight, 2015. 18(4): p. 28-30.
[71] de Weck, O.L., Fundamentals of systems engineering. 2009.