Received: 8 June 2022 | Revised: 30 November 2022 | Accepted: 16 January 2023

DOI: 10.1002/sys.21660

REVIEW ARTICLE

WILEY

Detecting emergence in engineered systems: A literature review and synthesis approach

Rune Andre Haugen[1] | Nils-Olav Skeie[2] | Gerrit Muller[1] | Elisabet Syverud[1]

[1]University of South-Eastern Norway, Kongsberg, Norway

[2]University of South-Eastern Norway, Porsgrunn, Norway

Correspondence
Rune Andre Haugen
Email: rune.a.haugen@usn.no

Significance of the Paper for Researchers and Practitioners: Researchers have been unable to reach a consensus on a common definition of the concept of emergence, while practitioners have experienced difficulty in recognizing the presence of emergence within the development of highly complex systems. The significance of this literature review paper is to summarize and synthesize the central literature in the field of emergence, including how it is described, categorized, as well as practical applications to detect this phenomenon. The research discusses different approaches and methods to detect emergence that fit different complexity categories, as emergence and complexity are intertwined phenomena. The paper provides a search log and screening process to make the findings reproducible. Further, the paper includes a metadata analysis of relevant literature and the relations between these to build confidence in the study. Finally, the paper contributes to the body of knowledge by including the dynamic dimension of the observer to the emergence vs. complexity relation.

Abstract

Modern product development often generates systems of high complexity that are prone to emergent behavior. The industry has a need to establish better practices to detect inherent emergent behavior when engineering such systems. Philosophers and researchers have debated emergence throughout history, tracing to the time of the Greek philosopher Aristotle (384–322 B.C.) and current literature has both philosophical and practical examples of emergence in modern systems. In this review paper, we investigate the phenomenon of emergent behavior in engineered systems. Our aim is to describe emergence in engineered systems and propose methods to detect it, based on literature. Emergence is in general explained as dynamic behavior seen at macro level that cannot be traced back to the micro level. Emergence can be known or unknown in combination with positive or negative. We find that best practices to engineer complicated systems should contain a sensible suite of traditional approaches and methods, while best practices to engineer complex systems need extensions to this considering a new paradigm using incentives to guide system behavior rather than testing it up-front.

KEYWORDS

automation, complexity, emergence, emergent behavior, integration

1 | INTRODUCTION

1.1 | Background

Modern product development generates large-scale assemblies consisting of state-of-the-art technology applied in system-of-systems architectures. The multi-layer architecture, the system partitioning, and the technology development at component level drive the overall system complexity and makes the system and the system-of-systems prone to emergent behavior. We observe that industries in the defense, space, and aerospace domains, driven by stringent product requirements and long lifetime are designing systems of high complexity. These industries search for methods to detect emergence.[1] Controlling emergence is the key factor for both operational success and failure in system-of-systems.[2] The challenge for systems engineers is to predict and analyze emergent behavior, especially undesirable behavior, in systems-of-systems.[3] System testing is traditionally seen as the prime mechanism to control the overall system behavior, but in a system-of-systems, this traditional approach is insufficient.[4,5] Hessami and Karcanias[6] state that increasing complexity requires a higher degree

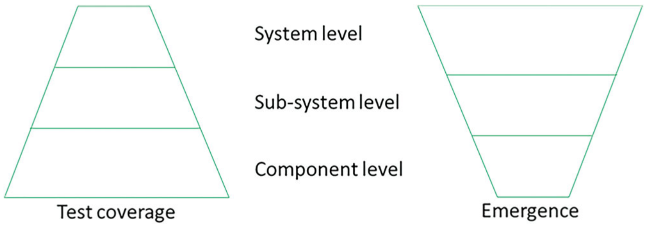

FIGURE 1 Principal sketch of the mismatch in test coverage and emergence.

of testing, and that integration approach and tools are critical for the success of engineering complex systems. In this paper, we relate system complexity to the four categories of simple, complicated, complex, and chaotic.[7]

System level testing in the defense, space, and aerospace industry typically lacks coverage due to the high cost of testing at this level. Therefore, complete system tests are rare in this type of industry. Close to real-system hardware test arenas are often used for real-time simulation, but even these are applied only to a limited degree because of the resource situation (people and money) and tight project schedules. While the system tests are expensive and resource consuming, the checks are often manual, time consuming, and require system expertise. Kjeldaas et al.[8] indicate that the product developers do not check all test data because it is too time consuming for busy system experts to manually check all test data. Hence, future product developments may not benefit from the knowledge available in the unchecked data. In addition, there is potential for unforeseen system behavior. Figure 1 shows this potential mismatch between the typical test coverage at different system levels and the emergence occurring at different system levels. This lack in test coverage at system level often leads to emergent behavior being discovered late in the development process or in worst case during the operational phase of the product.[8]

Philosophers and researchers have been debating the phenomenon of emergent behavior throughout history. The quote "*The whole is greater than the sum of its parts*" is related to Aristotle's (384–322 B.C.) quote "*The whole is something besides the parts*".[9] One plus one is greater than two due to interaction effects creating new behavior at higher system levels. Axelsson[10] gives a brief historical overview of research regarding emergence.

Emergence is commonly addressed into four specific categories, developed by Mittal and Rainey[11]: Simple, weak, strong, and spooky. Simple and weak emergence is readily predicted and reproduced in simplified models or simulations of the system. Contrary to weak emergence, strong emergence is not reproduceable by system simplification. Spooky emergence will not materialize in any model, not even models that simulates the complete system-of-systems in all details.[11]

In a series of papers on Emergence, McConnell[12–15] discuss the observer viewpoint and the importance of taking additional perspectives when searching for causes of emergence in systems.

Emergence and complexity are two sides to the same coin, representing degrees of difficulty for an observer to obtain a good understanding of the system and its behavior. This also indicates the

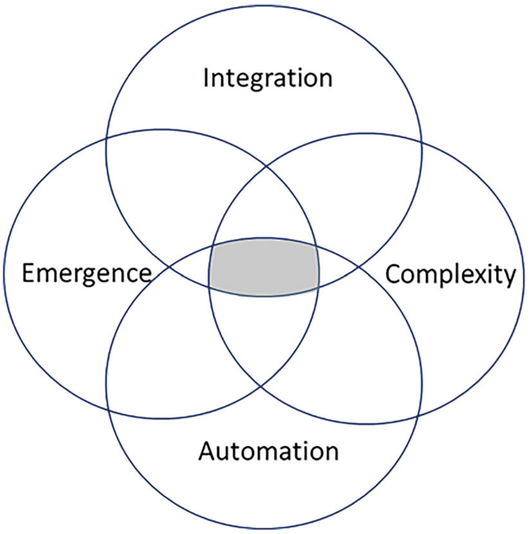

FIGURE 2 Principal sketch of areas of interest and their overlaps.

level of effort required to achieve that understanding, as well as what methods will be most useful in that endeavor. A sensible utilization of humans and machines becomes essential at higher degrees of emergence and complexity, and automation is then a critical measure to exploit the areas where machines outperform humans. As emergence shows in all levels of integration and more at higher levels, we need to focus our efforts at the highest level of integration to extract the main part of the non-intuitive inherent system behavior. Figure 2 visualizes the four areas of interest for our research case, also illustrating their overlapping areas. The key area to explore will be the area common to all four (gray area). *Emergence* is the main theme. According to Mittal et al.[16] *complexity* relates to emergence in the way that higher emergence categorized behavior typically arises in higher complexity categories requiring different measures to detect the behavior, if at all possible. *Automation* has the potential to improve the coverage of system analysis and test, and industrial companies have reported 15 ± 5 fold efficiency improvement with shifting the man versus machine task balance.[17] *Integration* is the product development domain where we focus our efforts in detecting emergence, which should start at the beginning of the project.

Both *system architecture & design* as well as *system integration* activities should make sure that most system errors- and undesired behaviors are detected. It is essential to balance the time used in these two above-mentioned activities with regards to process efficiency, and effort spend on Systems Engineering (SE) in total. According to Honour,[18] 15% of project cost spent on SE effort is optimal in the sense of keeping the actual cost- and schedule according to plan. However, Honour also found that the system complexity has an inverse effect on the total SE effort, where more complex systems tend to use less SE effort and relies more on "lower-level test-and-fix methods". Honor's research is for defense systems with up to 500 system-level requirements and up to 50 system-level external interfaces. Emergence is not considered in this work.

In this paper, we conduct a literature review to find typical descriptions of emergence, as well as means to detect emergent behavior. The review will focus on emergence as a phenomenon: how it is described, and which methods are proposed and applied to detect it. The goal of our review is to find strategies and methods that are known to detect emergent behavior in relevant engineered systems. We search for strategies that make it possible for the industry to detect emergence, in areas where the industry struggles with detecting it today.

1.2 | Contributions of the review paper

The contributions from this review will include:

- Emergence description and categorization: How the literature explains and categorizes emergence.
- Emergence relation to complexity: How types of emergences appear in types of engineered systems, and how this can depend on the observer.
- Identification of practical methods to detect emergence.

1.3 | Research questions

The literature review will answer the following research questions:

- Research question (RQ): What is the existing state of knowledge of detecting emergence in engineered systems?
- Sub-research question 1 (SRQ1): How does the literature explain and categorize emergence and its relation to complexity?
- Sub-research question 2 (SRQ2): What approaches and methods for detecting emergence are available from literature?

1.4 | Scope of survey

The remainder of this paper is structured as follows: Section on Methods presents methodology used for this literature search. Section on Results provides a review of selected literature that explains emergence in engineered systems and propose methods of detection, looking at differences and similarities in taxonomies and methods. Section on Discussion provides a discussion of the reviewed literature addressing the research questions and defines limitations of the study. Section on Conclusion concludes the paper, provides gained knowledge to detect emergence, and proposes future research.

2 | METHOD

We conducted a narrative synthesis literature review, which is part of the mixed methods review family.[19] Our goal was to get an overview of the research in the area of interest and how we can take advantage of this research using critical reflection in the synthesis process.[19]

We summarized the substance of the most relevant literature on emergence, specifically descriptions and methods for detection of this particular phenomenon. We investigated and debated the applied methodologies to see if there were any connections between methodologies used and the successful or unsuccessful outcomes.[20]

The authors established relevant taxonomies, descriptions, and methods by within our field of research. We generalized trends in the literature, to avoid an overwhelming level of details.[20] We allocated the literature into different directions, discussed these different directions, evaluated them, and concluded on the best way forward. We discussed the research outcomes found in this literature study in terms of the applicability to our continuing research.

The authors used a search strategy to crawl the titles, keywords, and abstracts of the databases *Web of Science*,[21] *Scopus*,[22] *IEEE Explore*,[23] *Wiley Online Library*,[24] and *Science Direct*[25] for defined search words. In addition, we used author keywords and keywords plus. We combined search words with Booleans, AND for different elements of the research question, and OR for different relevant synonyms. To include different wording in English and American, like behavior or behaviour, we used truncation (behavio?r). To include all forms of a word (emergent, emergence, emergentism), we used wildcard (emergen*). To restrict the search to specific wording, we used phrases ("Machine Learning"). To avoid restricting the search to specific domains we did not use proximity operators.

3 | RESULTS

3.1 | Literature search and selection

The literature search results can be found in Table A1 in *Appendix A: Literature Search Results*, which lists the primary sources found through the databases *Web of Science*,[21] *Scopus*,[22] *IEEE Explore*,[23] *Wiley Online Library*,[24] and *Science Direct*.[25] The authors retrieved the documents through the University of South-Eastern Norway (USN) library database.[26]

The selection criteria we used to filter out the most relevant literature to our research were the following:

- Focus on emergent behavior, not other types of emergent properties
- Focus on methods to detect emergent behaviors, not specific tools
- Focus on engineering, not other fields

The literature search log can be found in Table B1 in *Appendix B: Literature Search Log*, including more details regarding the searches and findings.

The aim was to select literature fitting into our area of interest, ref. Figure 2. Figure 3 shows the selection process, including three steps of screening:

- First, we scanned through the titles to exclude less relevant topics to our research, resulting in a down-selection from 756 to 101 records.

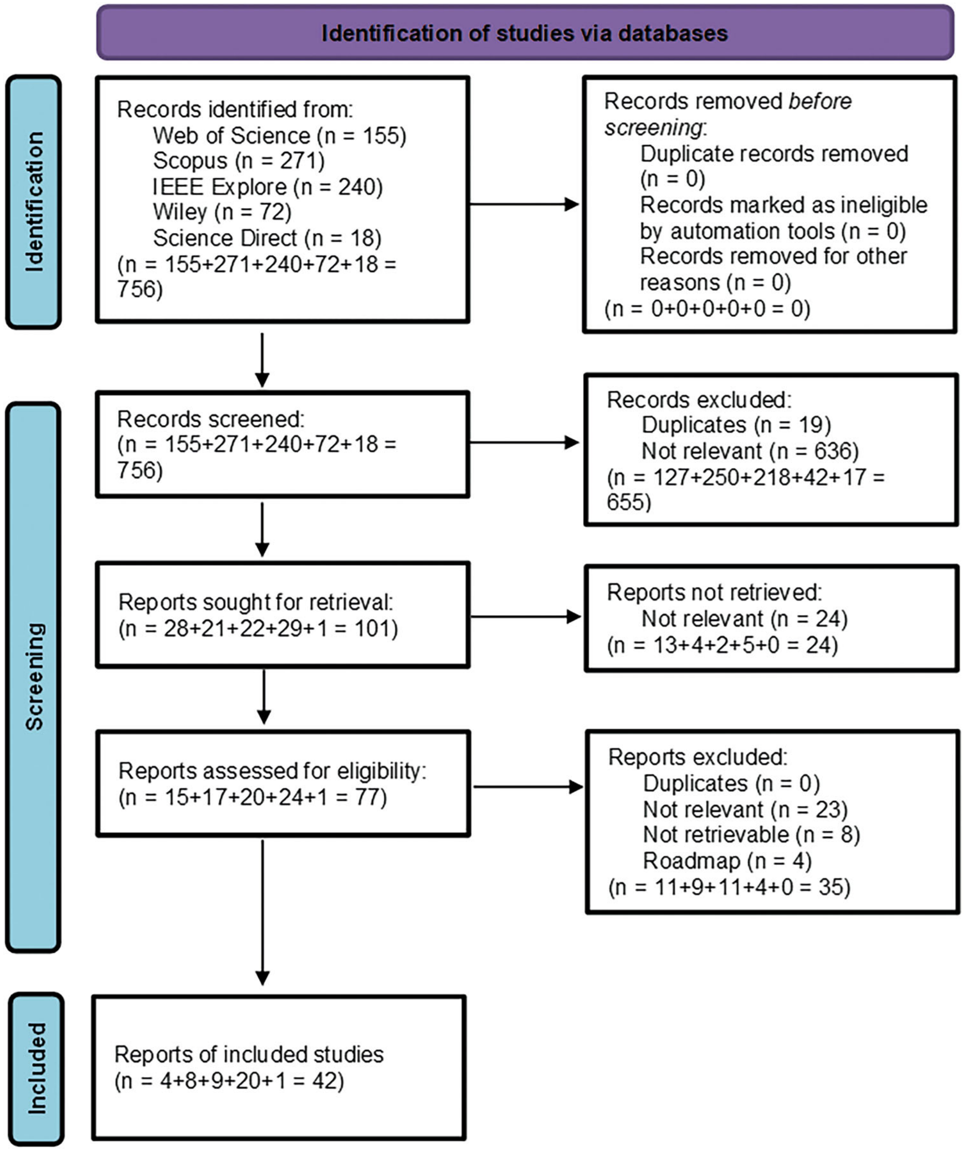

FIGURE 3 Literature search selection of primary references.

- Second, we read the abstracts to remove uninteresting research, resulting in a further down-selection from 101 to 77 records.
- Third, we skimmed through the text of the remaining articles to filter out even more not being sufficiently relevant to our research, resulting in a final down-selection from 77 to 42 records.

Figure 3 shows the PRISMA flow diagram[27] of the selection process of the primary references found through the database searches. Based on this search and selection process, we found central or pivotal literature within our field of study. This process was not sufficient for us to be able to generalize. Additional literature of interest was found by looking at the reference lists of the selected sources, using the so-called "snowballing effect".[19] This, to include sufficient literature in the field of study to be able to generalize with more confidence. See Table A2 in *Appendix A: Literature Search Results* for an overview of the secondary sources found through the "snowballing effect" from the primary sources, which we found to support the relevant information in the primary sources. We included 30 new records in this second round of literature selection. These secondary references completed the survey results, ending with an overall count of 72 records that the authors explored. Additional 10 references, including databases used for searching and literature review theory, completed the paper's reference list of 82 records.

3.2 | Literature metadata-analysis and mapping

We conducted a metadata-analysis of the selected references to ensure their validity by investigating their affiliations, publishing channels, and citations. The metadata-analysis is part of Table A1 and Table A2 in *Appendix A: Literature Search Results*.

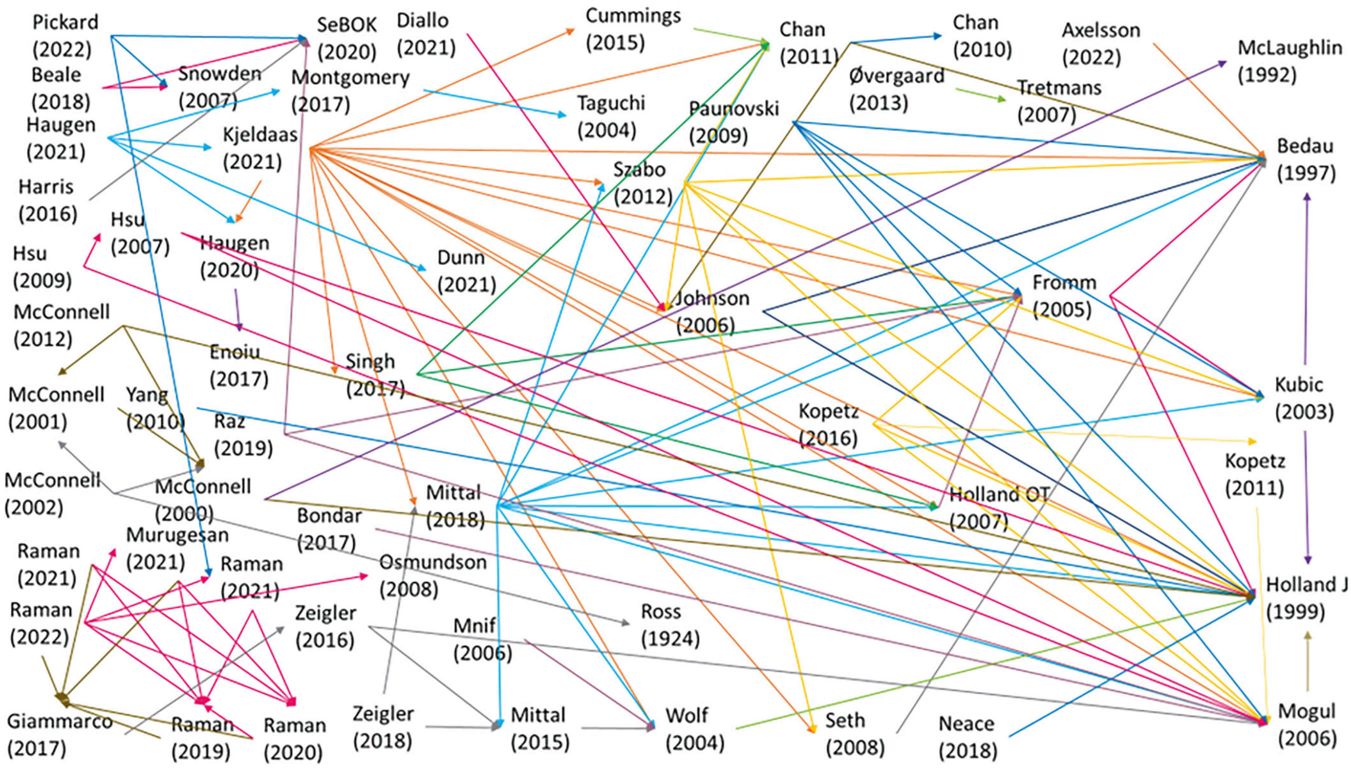

FIGURE 4 Mapping of relevant literature.

Figure 4 maps the literature selected for further study to see which citation relations there are among these (the color coding is for visualization purposes to see the citations more easily from different authors). Kjeldaas et al.,[8] Mittal et al.,[16] and Szabo and Teo[28] are the sources with the most references to others, containing information from many of the sources of older date. Bedau,[29] Holland,[30] and Mogul[31] are the sources that most of the others are referencing, which we then consider to be of high relevance.

3.3 | Emergence descriptions

Despite having been discussed for some 2500 years, general agreement on key characteristics of emergence is still lacking.[10] The description of emergence in literature is still fairly consistent.

Emergence can only occur at a system level: Axelsson[10] describes emergence as phenomena that occur only at a system level, while Fromm[32] defines a property to be emergent only if it appears on a higher level of complexity or organization. The patterns of emergence are only observed at a global level.[33] Kopetz[34] explains that "*Emergent properties are irreducible, holistic, and novel–they disappear when the system is partitioned into its subsystems.*"

Emergence is dependent on the context[30] and give rise to unique global properties at the system level.[34] Wolf and Holvoet[35] recommend to study emergence in the context in which the behavior is found.

Emergence implies interaction among systems and can also come from the lack of interaction among systems.[36] The patterns seen at higher level arise solely from "*interactions among lower-level components acting on rules which are executed using only local information without reference to the global pattern*".[33] Although the interactions are nonlinear, Holland[30] claims that it is possible "*to reduce the behavior of the whole to the lawful behavior of its parts, if we take the nonlinear interactions into account.*"

Callister and Andersson[37] use the related term "technical debt", which could result in emergence caused by many small shortcuts during the architecture and design phase.

3.4 | Emergence directions

Emergence appears in three different forms in literature: unpredictable form only, predictable form only, and both predictable and unpredictable forms. Table 1 shows some representative references from the literature connected to descriptions of the three emergence directions.

McLaughlin[38] is the earliest reference to emergence we have used in this survey and is the only one we have found claiming that emergence is purely unpredictable. We therefore assess the claim of emergence being purely of unpredictable character to be an outdated statement. The next direction to emergence we have used in this survey came with Bedau[29] claiming that emergence is purely

TABLE 1 Representative descriptions of emergence directions from relevant literature.

Reference	Descriptions of the emergence directions
McLaughlin[38]	Unpredictable form only: Emergence is connected to unpredictability, dismissing predictable forms of emergence by stating that a property is either reducible or emergent. Unpredictability does not equal emergence, since not all that is unpredictable is emergent. We can see physics, chemistry, and biology as sub-systems. When nature integrates these sub-systems into a system, emergent behavior will show. Scientific advances have reduced the domain of emergentism.
Bedau,[29] Mnif and Muller-Schloer[39]	Predictable form only: Strong (unpredictable) emergence is possible, but at the same time uncomfortably like magic and getting something out of nothing. Weak (predictable) emergence can explain autonomous behavior at macro level due to micro level dynamics. Complex systems exhibit weak emergence, and emergence can be both foreseen and unforeseen.
Fromm,[32] Holland,[33] Mittal and Rainey,[11] Mittal et al.,[16] Seth[81]	Predictable and unpredictable forms: Emergence can be divided into a deterministic and a stochastic region, deterministic being predictable and stochastic being unpredictable. Strong emergence is just outside the bounds of deterministic systems, entering the stochastic region. In this region, with the help of available knowledge, the stochasticity can be controlled. Once new knowledge about the novel behavior is obtained the system now portrays weak emergent behavior since the behavior is not novel anymore.

predictable. Bedau[29] is cited in literature of newer date like Mnif and Muller-Schloer,[39] but we haven't found any recent literature supporting this claim. We therefore assess the claim of emergence being purely predictable to be an outdated statement. The dominating direction in our field of study today is the third direction, introduced to our research in Fromm,[32] including both predictable and unpredictable types of emergence. This direction is confirmed in research of newer dates like Mittal et al.[16] and Mittal and Rainey.[11] We therefore assess the claim of emergence being both predictable and unpredictable to be the most common understanding of the phenomenon today, which most new research within our field build upon.

Kopetz et al.[40] bring another view to predicting emergence: "*We typically exploit expected beneficial emergence for intended product performance, while unexpected beneficial emergence can result in the product performing better than that for which it was designed. Expected detrimental emergence typically is allowed weaknesses of a product, while unexpected detrimental emergence could lead to unpredictable system failure*".[40] Johnson,[41] as well as Zeigler,[42] supports this distinction between beneficial (positive) and detrimental (negative) emergence. Therefore, we assess the categorization of emergence as positive or negative in combination with expected or unexpected as valid for the current research of emergence.

3.4.1 | Observer influence

Kopetz et al.[40] and Mittal et al.,[16] although agreeing on the direction of emergence, differ in how the observer influences the definition of emergence. Mittal et al.[16] include the observer in the definition of emergence by looking at how the phenomenon is experienced by the observer. Kopetz et al.[40] on the other hand excludes the observer from the definition of emergence, saying the phenomenon is what it is independent of the observer. We have not found any clear and common direction in the literature when it comes to the observer's impact on emergence, whether to look at the phenomenon subjectively through the eyes of the observer or purely objectively from a definition of a viewpoint. We therefore assess the statements of both Mittal et al.[16] and Kopetz et al.[40] regarding influence of the observer on emergence to be true, dependent on your point of view being subjective or objective. Donaldson[43] advocates the significant impact humans have on systems, while McConnell[15] claims the human impact to be highly dependent on the individual. Axelsson[10] states that the inclusion of an explicit observer is essential for understanding and handling emergence. In addition, the more an observer knows and understands the less strong the emergent properties become.

From the Systems Engineering Body of Knowledge (SEBoK)[44] we see that sufficient operational experience is necessary to put us in a position to be able to exploit positive- and avoid negative emergence, and that preventive actions alone are not enough. "*True hindsight and understanding comes from building multiple systems of the same type and deploying them, then observing their emergent behavior in operation and the side effects of placing them in their environments. If those observations are done systematically, and the emergence and side effects are distilled and captured in relation to the design of the systems—including the variations in those designs—and made available to the community, then we are in a position to predict and exploit the emergence.*[44]"

3.5 | Emergence and complexity

Emergence and complexity are highly related terms, and this coupling is necessary to understand when researching the emergence phenomenon.

The definition of complexity is inconsistent and often confused both in literature and practice.[45] Kopetz[34] classifies a system as complex only when the rational capabilities of the human mind cannot develop a set of models of adequate simplicity. Axelsson[10] associates complexity with the amount of information required to describe it in sufficient details.

Complex systems behavior is not reproducible as stated by Mittal et al.[16] "*the cause-and effect relations of complex systems are only coherent in retrospect and usually do not repeat.*"

Pickard and Beasley[46] see complicated systems as ordered, where cause and effect exists, although only seen by experts. On the contrary, cause and effect do not exist in complex systems, the system is by itself adaptive.[46]

Freund et al.[47,48] suggests categorizing of systems complexity as a combination of structural, dynamic, and environmental complexity. In this paper, we are influenced by Mittal et al.[16] and the Cynefin framework,[7] that categorize system complexity in simple, complicated, complex, and chaotic systems: *Simple systems typically exhibit simple emergence, complicated systems typically exhibit weak emergence, complex systems can potentially exhibit strong emergence, and chaotic systems can potentially exhibit spooky emergence.*

Complexity has a strong coupling towards uncertainty and unpredictability, as discussed by Beale and Tryfonas.[45] Hence, emergence in complex systems is per definition not predictable and the detection of emergence in complex systems is not possible with modern methods. Making the system predictable is therefore essential to avoid emergence. The Cynefin framework[7] proposes ways to deal with different degrees of complexity in a system, and that analysis can help transitioning to a more favorable category.

Mittal et al.[16] describe how exploring the interface between complicated and complex may generate updated knowledge of the system behavior that benefits new observers and helps describe it in sufficient details. Mnif and Muller-Schloer[39] describe how the system order depends on subjective decisions or capabilities of the observer.

The common understanding of complexity in the literature is that it is a measure of how difficult it is to understand a system and see the cause-and-effect relations, which is very much aligned with the descriptions we see in the literature regarding emergence. Mnif and Muller-Schloer[39] have the same view on complexity as we see in Mittal et al.[16] with regards to emergence, being that it is dependent on the observer. Therefore, we see both complexity and emergence as subjective measures related to the difficulty of describing a system (complexity) and its behavior (emergence) and being two sides to the same coin.

3.6 | Emergence detection

"*We will never be able to solve all emergent misbehavior problems, especially as system complexity increases. However, we can and should be able to recognize recurring patterns of misbehavior, and to learn enough from experience to be able to avoid or repair many of the common patterns.*[31]"

The literature survey reveals a large number of methods to detect emergence. We explore three different approaches: test coverage, manual versus automatic task balance, and combined modeling and simulation efforts.

3.6.1 | Test coverage

It is often difficult to have a full test coverage during a system test. Even for systems where full test coverage is possible, it may be difficult to evaluate and assess the test results.[8] Methods to assess test coverage will help targeting the most important tests and thereby limit the efforts and maximizing the outcome of test.

Osmundson et al.[3] support design of experiments as a means for analyzing emergent behaviors to predict favorable and unfavorable consequences in order to architect systems to better assure desired results.

Khan and Jing[49] lay out a formal approach to study the emergent behaviors of a system through a method using temporal logic. Instead of exploring all of the possible system state space, which may grow exponentially, the temporal logic method helps in computing the polarization and momentum for generating the emergent behavior set, and accordingly from which the emergent behaviors can be detected.

Fractional factorial test methods help assess test coverage. Dunn[50] and Montgomery[51] describe different fractional factorial test methods as means to reduce the scope of testing. Taguchi et al.[52] describe a method for systematic testing through orthogonal arrays.

The design of experiments is an important approach to facilitate detection of emergent behaviors. Haugen and Ghaderi[53] applies two-level fractional factorial design of experiments together with Bayesian statistical inference to simulate and increase the prior knowledge of the probabilities of different emergent behaviors in an engineered system.

The authors find the fractional factorials and temporal logic methods to be good alternative means to explore the most interesting areas of the parameter state space regarding detection of emergence in the system of interest.

3.6.2 | Manual and automatic tasks

The system checks are often manual, time consuming and require system expertise. Methods that help in finding a balance between manual and automatic tasks can help limiting the efforts required by system experts. Haugen and Mansouri[17] explore an industrial test system with regards to the manual versus automatic task balance, using a systems thinking approach. McConnell[13] supports that the systems thinking approach can be useful to understand the emergent properties of a system.

Øvergaard and Muller[54] is a practical paper finding the advantage of automation in an industrial test campaign. Giammarco[36] and Szabo and Teo[28] are theoretical papers describing what is suitable for automation and not, supported by the formal analysis exploiting human and machine strengths in Harris and Narkevicius.[55]

A more sophisticated method like Machine Learning is used in today's state of the art research in emergence. Raz et al.,[56] Raman and Jeppu,[57–59] Raman et al.,[60] and Raman and Murugesan[61,62] are exploring in recent papers how Machine Learning algorithms can help detect emergent behavior. Although Machine Learning can be used for both automatic and manual tasks, these papers seem to agree that automation measures can be very helpful in ensuring sufficient test coverage and extraction of information from test data to facilitate detection of emergent behavior.

TABLE 2 Representative literature highlights on manual versus automatic task balance approach.

References	Literature highlights on emergence detection methods
Diallo et al.[79]	Suspicious values produced by the statistical debugging process can be manually inspected to determine emergent behavior.
Enoiu et al.,[80] Øvergaard and Muller[54]	We can reduce the test time used for manual testing by more than 90% by automating the test procedures that are suitable for automation.
Giammarco[36]	Automated tools with built in simulators become essential for verifying and validating behavior logic in a reasonable amount of time.
Haugen and Mansouri[17]	Automating test execution and test result analysis can remove bottlenecks in the test process.
Raman and Jeppu,[57–59] Raman et al.,[60] and Raman and Murugesan[61,62]	The Measures of Performance (MoP's) and Measures of Effectiveness (MoE's) can be monitored using Machine Learning to look for changes that give or could give raise to emergent behaviors. Machine Learning could be used to adapt system behavior in tandem with the evolution of emergent behavior in a complex system-of-systems.
Raz et al.[56]	Systems Engineering is the first step towards understanding and controlling emergent behavior, while Raz et al. assert that future methods will rely on Machine Learning methods.
Szabo and Teo[28]	A key challenge is the need for abstractions of the micro and macro levels, which are difficult to achieve in an automated manner, and hence most approaches rely on a post-mortem observation of the simulation by a system expert.

We assess that automation is a beneficial and necessary approach to detection of emergence in the system of interest, but only related to what is suitable for automation. Table 2 presents a summary of the representative findings from literature.

3.6.3 | Combined modeling and simulation efforts

The literature supports the use of combined modeling and simulation efforts to provide insight into the causes of observed, possibly emergent behavior.[1] Computer-based exploration can bring new insight.[30] A system-of- systems modeling and simulation framework architecture can provide identification and quantification of emergent behavior.[63] Such models are required to measure the existence, type, and level of emergent behavior of the system-of-systems.[64]

Kossiakoff et al.[65] advocates system behavior models where performance is analyzed at multiple levels (system, sub-system, and component). Paunovski et al.[66] states that although an emergent phenomenon is only visible at runtime operation and cannot be captured a priori with a model of the system, the conceptual models can be updated with the gained knowledge through simulations.

Neace and Chipkevich[67] suggests to use emergence requirements as a method for governance of emergence in a complex system.

Hyun et al.[68] and Shin et al.[69] use a fault database to incrementally build a framework to increase the detection of emergence, being an iterative process to increase a common understanding.

Yang et al.[70] states that "*Many emergent behaviors burst out in different simulations accompanying the changing parameters and different scenarios.*" This may lead researchers to limit their simulations such as only covering weak emergent behavior as they find these as the most interesting with respect to engineering applications.[71]

Pourafzal and Fereidunian[72] use three differential equations known as Lorenz equations in which adopting different parameters dictate the phase transition between order and chaos, meaning the transition between complex and chaotic systems as well as strong and spooky emergence.

Agent-Based Simulation (ABS) simulates the interactions of autonomous objects (called agents) to identify, explain, generate, and design emergent behaviors. The local interactions serve to create global structures and patterns of behavior.[2,73–75] The sources of emergence can be agent properties, inter-agent interactions, the influence of the environment on the agents' actions, and ongoing evolutionary processes on a part of the agents as well as the environment.[76]

There seems to be two different approaches in the literature when we are talking about modelling and simulation, one approach focusing on the design part (left side of the Vee-model) and the other focusing on the integration part (right side of the Vee-model). Both approaches are frequently supplied with new research, especially related to Model-Based Systems Engineering (MBSE) and Machine Learning (ML) methods. We assess both approaches to be equally important, and that they together form the Systems Engineering process to detect and understand different types of emergence. We believe emergence can be understood in the design aspect through modeling efforts, but we need testing on different realization levels (hardware in the loop) of the system to detect weak and higher categories of emergence.

The goal of model-based testing is to reduce the integration and test effort of industrial systems.[77] Model Based Systems Engineering (MBSE) can benefit from simulations for early validation of the system design.[77] Model Based Systems Integration (MBSI) extends the MBSE process by increasing the system integration impact early in the design.[78]

4 | DISCUSSION

The endeavor of studying and understanding the phenomenon of emergence has been researched throughout centuries and will keep being investigated for centuries to come.[9,10] The Systems Engineering process and methods are a good start for detecting emergence by incorporating working procedures making this possible.[56] Raz et al.[56] Raman and Jeppu.[57–59] Raman et al.[60] and Raman and Murugesan[61,62] state that modeling can help us regarding simple emergence, while simulation is required to detect weak emergence. We can update our modeling efforts when we understand the simulated emergent behavior, to evolve the system design and exploit and/or mitigate the detected emergence.

Several methods can be used within the Systems Engineering processes to detect emergent behavior. How the testing is set up, Design of Experiments (DoE), is crucial to facilitate detection of emergence by providing sufficient stimuli from component interactions, human input, and environmental impact to trigger the emergent behaviors of the system.[53] A system model with parameters related to the inputs and features of dependencies can help see the effect of changes and causes of emergence.[1] Modeling & simulation can detect emergence but depends on the correctness of the model.[66] Building a fault knowledgebase can help detecting emergence but will require building experience through an iterative learning process.[69] Statistical debugging is a software (SW) method that can help in detecting emergence.[79] Machine Learning (ML) is a method that can detect and predict potential future detection of emergence, but can also introduce emergence itself.[56]

The outcome of these findings for our ongoing research in detecting emergent behavior in engineered systems is that we should take advantage of existing methods like orthogonal arrays (fractional factorial design of experiments and Taguchi matrices) to systematically test what is relevant, as one of the paradigms to testing. Further, we should use modeling & simulation to provide system data. Finally, we need appropriate data analysis techniques (system model inspection, fault knowledgebase, statistical debugging, and Machine Learning) to extract relevant information from our system models and test data.

We want to identify and take advantage of the positive emergence, while detect and mitigate negative emergence. Many methods have shown promising results in different case studies.[1–3,17,28,30,36,49–54,56–67,70,71,73–80] The approaches range from intuitive specialized model views used by Guariniello et al.[1] to advanced non-intuitive methods like Machine Learning used by Raz et al.[56] Different approaches will serve different cases to a varying degree, requiring a thorough evaluation process to select the most appropriate method to utilize in each case. In the following, we will discuss the proposed research questions.

4.1 | SRQ1: How does the literature explain and categorize emergence and its relation to complexity?

The phenomenon of emergence has many descriptions, but they all roughly point in the same direction.[10,30,32–36] The main idea is that the macro level exposes behavior that cannot be traced directly back to the micro level. Different sources categorize emergence into defined forms and types, although they dispute the applicability of these and the impact of the observer. There are three main directions found in existing literature on emergence, being (1) unpredictable,[38] (2) predictable,[29,39] and (3) both.[11,16,32,33,81] The unpredictable direction covers strong and spooky types of emergence,[16] while the predictable direction covers simple and weak types of emergence.[16]

For our ongoing research, we describe emergence as behavior at macro level that cannot be traced directly down to micro level. We categorize emergence as predictable (simple and weak) as well as unpredictable (strong and spooky), and we look at it subjectively from an observer point of view.

4.2 | SRQ2: What approaches and methods for detecting emergence are available from literature?

There are several methods within three main approaches for detecting emergence. The first approach is to focus on what we test to establish a reasonable test coverage.[53] Orthogonal arrays is a method used to systematically test to a given interaction level,[50–52] and using temporal logic is another related method.[49] The second approach is related to how efficient we are able to execute a test cycle, looking into the manual versus automatic task balance.[17,28,36,54,80] Machine Learning is a method of interest to automate data analysis,[56–62] and statistical debugging is another.[79] The third approach is modeling and simulation to increase our level of understanding through simulation and updated models.[30,63,64,67,70] MBSE is an overarching method being used,[77] as well as MBSI.[78] More specific methods used in modeling are system behavior modeling,[65] system interface modeling,[1] and conceptual modeling.[66] More specific methods used in simulation are model-based testing[82] and agent based simulation.[2,3,71,73–76]

Mogul[31] claims we will never be able to solve all emergent behavior of systems. This claim is reasonable, especially when we are dealing with complex systems and/or strong emergence, leading us into a potential infinite endeavor in pursuing all emergent behavior.

The effectiveness of detecting different types of emergent behavior through simulations will vary. Simple emergence can be managed through the system model,[16] and does not need to be simulated. Strong and spooky emergence are unpredictable and do not usually repeat,[16] meaning that the simulation effort may not be worthwhile. Weak emergence is then the type of emergence that can be found most effectively through simulation. Modeling & simulation cannot detect all emergent behaviors in the first iteration cycle, as we do not yet possess all necessary information.[66] We need additional iteration cycles to uncover more and more of the remaining emergent behaviors inherent in the system not revealed through previous iterations.[66] Systems architects can guide emergence through adapting and accommodating changes in requirements and technology, in a timely manner (see Hsu et al.[4,5] for plenum discussions on if and how we can engineer emergence).

For our continuing research, we should focus on how to set up the required test suite to trigger the emergent behaviors of the system. In this process, we must evaluate the need for different types of testing like virtual-, real-world-, hardware-, software-, and stress testing. This is a natural first step, being a prerequisite for any value in later data analysis. Further, we should focus on data from modeling and simulation

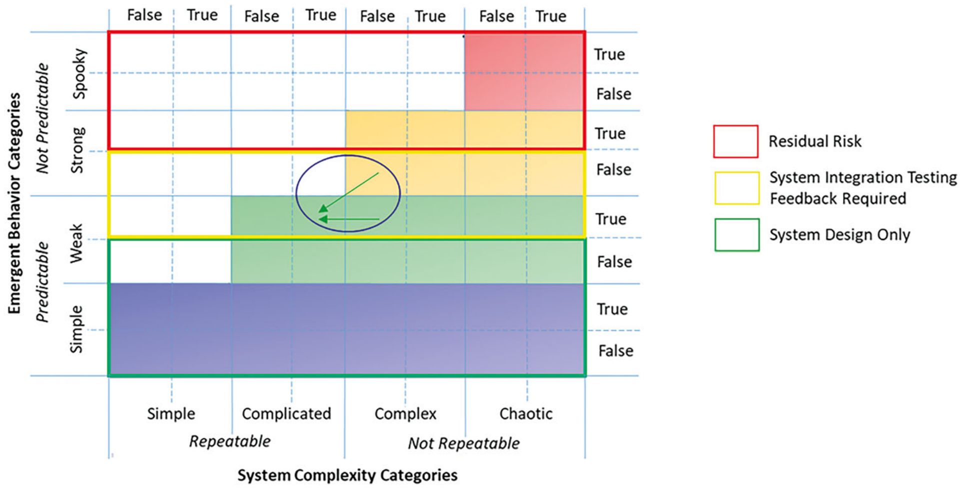

FIGURE 5 Relations and dynamics in emergence and complexity categorization.

iterative cycles to increase the likelihood of detecting as much as possible of the system emergence. We need to keep in mind that the capabilities of both the test arenas being used and the observers interpreting the results are impacting the subjectivity in the emergence categorization.

4.3 | *RQ*: What is the existing state of knowledge of detecting emergence in engineered systems?

The current research coverage in the area of emergence stops in the transition between weak and strong type of emergence, weak emergence being predictable while strong emergence being unpredictable.[1,2,17,28,30,36,49–54,56–67,70,71,73–80] No strategies or methods have proven to be successful to detect strong emergence.[16] The literature is looking at simple and weak emergence in complicated and complex systems.[1,8,57–62,66,69,79] The perceived emergence and complexity are dependent on the observer.[11,16,39] The observer may see the emergence as strong at first, not understanding the emerging system behavior. After obtaining more knowledge through test and analysis, the observer may perceive the emergence as weak, understanding the emerging system behavior. The same observer impact applies for complexity categories, as in the transition from a perceived complex system to a complicated system.[39]

The ongoing research indicates that we are moving into a shift of paradigm when it comes to how we deal with emergent behavior in complicated versus complex systems. For a complex system, we might need to move away from the traditional test coverage approach and look more into incentives to adapt the system behavior.[4,5,61] No amount of testing can guarantee that a complex system will not fail, as well as the number of permutations rendering the calculations and storage impossible.[55] Complicated systems can be seen as sub-systems of a complex system-of-systems, and are the bedrock of Systems Engineering practice.[46]

This literature study is limited in the sense of being a narrative synthesis and not a systematic review, which may result in not including all relevant literature on emergence.

In this paper, we postulate that it is possible to build on existing practices and methods to detect true weak and false strong emergence in engineered true complicated and false complex systems. False strong emergence is the part of strong emergence that has to do with the observer's perception due to lack of system knowledge. This will again help us to develop best practices for emergence detection further. By this false and true strong emergence separation, we are able to push the boundary of emergence detection to also include the part of perceived strong emergence that really is weak emergence. The transitions in the yellow region in Figure 5 show the part of emergence that could benefit both academia and industry to do more evolutionary research.

The direction for our continuing research will lay in the transition between predictable weak emergence and unpredictable strong emergence in combination with the transition between repeatable complicated systems and non-repeatable complex systems, see the purple circle in Figure 5. We will look into how the Systems Engineering process and methods can help the observer to increase knowledge sufficient to make the transition from perceived strong emergence in a perceived complex system to what is actually weak emergence in a complicated system. The aim is to create guidelines to facilitate the necessary increase in knowledge through simulation, including how to set up the test suite and how to perform the appropriate data analysis.

5 | CONCLUSION

The industry developing engineered systems typically has problems detecting emergent system behavior. These systems are designed to exhibit certain known and desired emergent behavior, but unfortunately also show some unknown and undesired behavior. The industry has a need to find better practices to detect more of this emergence in engineered systems.

Emergent behavior is novel macro-level behavior that cannot be traced directly down to the micro level parts. Interactions between the micro-level parts cause new behavior at macro level. Emergent behavior can be seen as a scale where increasing difficulty to predict and understand the macro level behavior indicates a higher degree of emergence. We need several approaches, methods, and tools, as well as much knowledge, to detect and understand the different levels of emergence. Increased observer knowledge can help to move the subjectively experienced macro level behavior down the emergence scale. We typically want to detect the inherent emergent behavior of the system under test to be able to utilize the beneficial emergence and mitigate the detrimental emergence.

Emergence is highly correlated with complexity. Complexity can be seen as a scale where increasing difficulty to describe and understand the system indicates a higher degree of complexity. Increased observer knowledge can help to move the subjectively experienced complexity down this scale. An interesting region on this scale is the transition between complicated and complex systems. Typically, a complex system can be seen as a system-of-systems consisting of complicated systems. A complicated system is difficult but possible to control, while a complex system is not controllable. Best practices to engineer complicated systems should contain a sensible suite of traditional approaches and methods, while best practices to engineer complex systems need extensions to this considering a new paradigm using incentives to guide system behavior rather than testing it up-front.

Further research is needed to evaluate what approaches and methods that work to what degree to detect weak, including subjectively strong emergence, in engineered complicated, including subjectively complex systems. Based on this output, we will be able to create best practices for that purpose. In addition, further research is needed to measure how effective the above set of best practices is for complex systems, as well as how additional steps will take effect. The Systems Engineering community needs continuous evolutionary research on emergence and complexity, and the research front is looking at weak emergence in complicated and complex systems.

ACKNOWLEDGMENTS

We would like to acknowledge the Human Systems Engineering Innovation Framework (H-SEIF2) project, the Norwegian Industrial Systems Engineering (NISE), -and Self-Monitoring, Analysis, and Reporting Technology (SMART) Research Groups at the University of South-Eastern Norway for their contributions, providing valuable feedback to our proposed work during the survey period. In addition, we would like to acknowledge the Research Council of Norway for their funding. Finally, we would like to acknowledge the KONGSBERG company for sharing relevant industry information through their subject matter experts and databases.

DATA AVAILABILITY STATEMENT

Data sharing is not applicable to this article as no new data were created or analyzed in this study

ORCID

Rune Andre Haugen https://orcid.org/0000-0002-5567-3179
Nils-Olav Skeie https://orcid.org/0000-0001-6358-9308
Gerrit Muller https://orcid.org/0000-0003-1952-8377
Elisabet Syverud https://orcid.org/0000-0002-6041-071X

REFERENCES

1. Guariniello C, Khalid RA, Fang Z, Delaurentis D. System-of-systems tools and techniques for the analysis of cyber-physical systems. *Sys Eng*. 2020;23(4):480-491. doi:10.1002/sys.21539
2. Hsu J, Butterfield M. Modeling emergent behavior for systems-of-systems. *INCOSE Int Symp*. 2007;17(1):1811-1821. doi:10.1002/j.2334-5837.2007.tb02985.x
3. Osmundson JS, Huynh TV, Langford GO. KR14 emergent behavior in systems of systems. *INCOSE Int Symp*. 2008;18(1):1557-1568. doi:10.1002/j.2334-5837.2008.tb00900.x
4. Hsu J, Axelband E, Madni A, Dagli C, McKinney D. 3.5.0 can we engineer the emergent behavior of a system-of-systems? *INCOSE Int Symp*. 2009;19(1):578-592. doi:10.1002/j.2334-5837.2009.tb00969.x
5. Hsu J, Axelband E, Rouse B, Madni A, Sheard S. 7.6.0 How to engineer the emergent behavior of a system of systems. *INCOSE Int Symp*. 2008;18(1):895-908. doi:10.1002/j.2334-5837.2008.tb00850.x
6. Hessami AG, Karcanias N. 9.6.2 Integration of operations in process systems: complexity and emergent properties. *INCOSE Int Symp*. 2011;21(1):1153-1158. doi:10.1002/j.2334-5837.2011.tb01274.x
7. Snowden D, Cognitive Edge. Accessed 25th November, 2021. https://www.cognitive-edge.com/
8. Kjeldaas KA, Haugen RA, Syverud E. *Challenges in detecting emergent behavior in system testing*. Wiley; 2021:1211-1228.
9. Ross WD. *Aristotle's metaphysics: A revised text with introduction and commentary*. Clarendon Press; 1924.
10. Axelsson J. What systems engineers should know about emergence. *INCOSE Int Symp*. 2022;32(1):1070-1084. doi:10.1002/iis2.12982
11. Mittal S, Rainey L. *Harnessing emergence: the control and design and emergent behavior in system of systems engineering*. 2015.
12. McConnell GR. 4.2.1 Emergence: A challenge for the systematic. *INCOSE Int Symp*. 2000;10(1):202-207. doi:10.1002/j.2334-5837.2000.tb00377.x
13. McConnell GR. 3.1.4 Emergence: A partial history of systems thinking. *INCOSE Int Symp*. 2002;12(1):90-98. doi:10.1002/j.2334-5837.2002.tb02447.x
14. McConnell GR. 6.3.3 Emergence: Open your eyes to new vistas. *INCOSE Int Symp*. 2001;11(1):489-497. doi:10.1002/j.2334-5837.2001.tb02332.x
15. McConnell GR. 5.3.1 Emergence: All in the minds. *INCOSE Int Symp*. 2012;22(1):705-717. doi:10.1002/j.2334-5837.2012.tb01366.x
16. Mittal S, Diallo S, Tolk A. *emergent behavior in complex systems engineering: A modeling and simulation approach*. Wiley; 2018:1-395.
17. Haugen RA, Mansouri M. *Applying systems thinking to frame and explore a test system for product verification; A case study in large defence projects*. Wiley; 2020:78-93.
18. Honour E, *Systems Engineering Return on Investment*. University of South Wales; 2013. https://www.hcode.com/seroi/documents/SE-ROI%20Thesis-distrib.pdf
19. Sutton A, Clowes M, Preston L, Booth A. Meeting the review family: exploring review types and associated information retrieval requirements. 2019:202-222. doi:10.1111/hir.12276
20. Randolph J, A guide to writing the dissertation literature review. Practical Assessment, Research, and Evaluation; 2009.
21. Clarivate. Web of Science. 2021. https://clarivate.com/webofsciencegroup/solutions/web-of-science/
22. Elsevier. Scopus. 2021. https://www.scopus.com/home.uri
23. IEEE. IEEE Explore. 2021. https://ieeexplore.ieee.org/Xplore/home.jsp
24. Wiley. Wiley Online Library. 2022. https://onlinelibrary.wiley.com/

25. Elsevier. Science Direct. 2022. https://www.sciencedirect.com/
26. Unit. ORIA. 2021. https://bibsys-almaprimo.hosted.exlibrisgroup.com/primo-explore/search?vid=HIT
27. Page MJ, McKenzie JE, Bossuyt PM, et al. The PRISMA 2020 statement: An updated guideline for reporting systematic reviews. *BMJ*. 2021;372:n71. doi:10.1136/bmj.n71
28. Szabo C, Teo Y, An integrated approach for the validation of emergence in component-based simulation models. 2012:2739-2749.
29. Bedau MA. Weak emergence. *Nous*. 1997;31(s11):375-399. doi:10.1111/0029-4624.31.s11.17
30. Holland J. Emergence. *From Chaos to Order*. 1999.
31. Mogul JC. Emergent (mis)behavior vs. complex software systems. *Association for Computing MachineryNew YorkNYUnited States*. 2006:293-304.
32. Fromm J, Types and forms of emergence. arXiv: Adaptation and Self-Organizing Systems. 2005;
33. Holland OT, Taxonomy for the modeling and simulation of emergent behavior systems. In: *Agent Directed Simulation Symposium, ADS 2007 - Proceedings of the 2007 Spring Simulation Multiconference, SpringSim 2007*. Association for Computing Machinery, Inc; 2007:28-35.
34. Kopetz H. Emergence. Real-time systems. *Real-time systems series*. 2 ed. Springer; 2011:43-45. chap 2.4.
35. Wolf T, Holvoet T. *Emergence versus self-organization: different concepts but promising when combined*. Springer; 2004:1-15.
36. Giammarco K, Practical modeling concepts for engineering emergence in systems of systems. 2017:1-6.
37. Callister PS, Andersson J. *Evaluation of System Integration and Qualification Strategies using the Technical Debt metaphor; a case study in Subsea System Development*. Wiley; 2015.
38. McLaughlin B. The rise and fall of British emergentism. In: Beckermann A, Flohr H, Kim J (eds.) *Emergence or reduction? Essays on the prospects of nonreductive physicalism*. De Gruyter; 1992:49-93.
39. Mnif M, Muller-Schloer C, Quantitative Emergence. 2006:78-84.
40. Kopetz H, Bondavalli A, Brancati F, Frömel B, Höftberger O, Iacob S. Emergence in Cyber-Physical Systems-of-Systems (CPSoSs). In: Bondavalli A, Bouchenak S, Kopetz H, (eds.) *Cyber-physical systems of systems: Foundations – A conceptual model and some derivations: The AMADEOS legacy*. Springer International Publishing; 2016:73-96.
41. Johnson CW. What are emergent properties and how do they affect the engineering of complex systems? *Reliability Engineering and System Safety*. 2006;9(12):1475-1481. doi:10.1016/j.ress.2006.01.008
42. Zeigler BP. A note on promoting positive emergence and managing negative emergence in systems of systems. *The Journal of Defense Modeling and Simulation*. 2016;13(1):133-136. doi:10.1177/1548512915620580
43. Donaldson W. In praise of the "Ologies": A discussion of and framework for using soft skills to sense and influence emergent behaviors in sociotechnical systems. *Systems Engineering*. 2017;20(5):467-478. doi:10.1002/sys.21408
44. Adcock R, Jackson S, Fairley D, Singer J, Hybertson D, Cloutier RJ, ed. "Emergence" in The Guide to the Systems Engineering Body of Knowledge (SEBoK). 2021. http://www.sebokwiki.org
45. Beale D, Tryfonas T. Exploration of the complex ontology. *INCOSE Int Symp*. 2018;28(1):1549-1563. doi:10.1002/j.2334-5837.2018.00567.x
46. Pickard AC, Beasley R. Engineering complicated systems still needs systems engineering and thinking. *INCOSE Int Symp*. 2022;32(1):721-736. http://doi.org/10.1002/iis2.12960
47. Freund L, Al-Majeed S, Millard A, Towards the definition of a strategic complexity management framework for complex industrial systems. 2021.
48. Freund L, Al-Majeed S, Millard A, Case studies key-findings of a strategic complexity management framework for industrial manufacturing systems. 2021.
49. Khan TA, Wang J. On formalization of emergent behaviors in multiagent systems with limited interactions. 2016:0553-0558.
50. Dunn K, Process improvements using data. 2021.
51. Montgomery DC. *Design and analysis of experiments*. 8th ed., Wiley; 2017.
52. Taguchi G, Jugulum R, Taguchi S. *Computer-based robust engineering: Essentials for DFSS*. ASQ Quality Press; 2004.
53. Haugen RA, Ghaderi A, Modelling and simulation of detection rates of emergent behaviors in system integration test regimes. Vitenskapelig artikkel. *Linköping Electronic Conference Proceedings*. 2021;
54. Øvergaard A, Muller G. *System verification by automatic testing*. Wiley; 2013:356-367.
55. Harris SD, Narkevicius JM. Emergent failure modes and what to do about them. *INCOSE Int Symp*. 2016;26(1):1044-1058. doi:10.1002/j.2334-5837.2016.00210.x
56. Raz AK, Llinas J, Mittu R, Lawless W, Engineering for emergence in information fusion systems: A review of some challenges. In: *FUSION 2019 - 22nd International Conference on Information Fusion*. Institute of Electrical and Electronics Engineers Inc.; 2019.
57. Raman R, Jeppu Y, Formal validation of emergent behavior in a machine learning based collision avoidance system. In: *SYSCON 2020 - 14th Annual IEEE International Systems Conference, Proceedings*. Institute of Electrical and Electronics Engineers Inc.; 2020.
58. Raman R, Jeppu Y, Does the complex SoS have negative emergent behavior? Looking for violations formally. In: *15th Annual IEEE International Systems Conference, SysCon 2021 - Proceedings*. Institute of Electrical and Electronics Engineers Inc.; 2021.
59. Raman R, Jeppu Y, An approach for formal verification of machine learning based complex systems. 2019:544-559.
60. Raman R, Gupta N, Jeppu Y. *Framework for formal verification of machine learning based complex system-of-system*. Wiley; 2021.
61. Raman R, Murugesan A. Framework for complex SoS emergent behavior evolution using deep reinforcement learning. *INCOSE Int Symp*. 2022;32(1):809-823. doi:10.1002/iis2.12965
62. Murugesan A, Raman R, Reinforcement learning for emergent behavior evolution in complex system-of-systems. IARIA; 2021.
63. Cummings MA, Identifying and quantifying emergent behavior through system of systems modeling and simulation. 2015. https://calhoun.nps.edu/handle/10945/47243
64. Bondar S, Hsu JC, Pfouga A, Stjepandić J. Agile digital transformation of system-of-systems architecture models using Zachman framework. *J Ind Inf Integr*. Sep 2017;7:33-43. doi:10.1016/j.jii.2017.03.001
65. Kossiakoff A, Flanigan DA, Seymour SJ, Biemer MS. Systems engineering principles and practice. *Systems Engineering & Management*. 3rd ed.. Wiley; 2020.
66. Paunovski O, Eleftherakis G, Cowling AJ. Disciplined exploration of emergence using multi-agent simulation framework. *Article Comput Inf*. 2009;28(3):369-391.
67. Neace KS, Chipkevich MBA, Designed complex adaptive systems exhibiting weak emergence. In: *Proceedings of the IEEE National Aerospace Electronics Conference, NAECON*. Institute of Electrical and Electronics Engineers Inc.; 2018:214-221.
68. Hyun S, Song J, Shin S, Baek YM, Bae DH, Pattern-based analysis of interaction failures in systems-of-systems: A case study on platooning. In: *2020 27th Asia-Pacific Software Engineering Conference (APSEC)*. 2020:326-335.
69. Shin S, Hyun S, Shin Y, Song J, Bae D, Uncertainty based fault type identification for fault knowledge base generation in system of systems. 2021.
70. Yang KW, Chen YW, Lu YJ, Zhao QS, The study of guided emergent behavior in system of systems requirement analysis. In: *2010 5th International Conference on System of Systems Engineering*, SoSE 2010. 2010.
71. Singh S, Lu S, Kokar MM, Kogut PA, Martin L, Detection and classification of emergent behaviors using multi-agent simulation framework (WIP). Society for Modeling and Simulation International (SCS); 2017.
72. Pourafzal A, Fereidunian A, A complex systems approach to feature extraction for chaotic behavior recognition. 2020:1-6.

73. Chan WKV, Son Y, Macal CM, Agent-based simulation tutorial - simulation of emergent behavior and differences between agent-based simulation and discrete-event simulation. 2010:135-150.
74. Chan WKV, Interaction metric of emergent behaviors in agent-based simulation. 2011:357-368.
75. Hsu J, Clymer J, Garcia Jr J, E Gonzalez. Agent-based modeling the emergent behavior of a system-of-systems. *INCOSE Int Symp*. 2009;19(1):1581-1590. doi:10.1002/j.2334-5837.2009.tb01036.x
76. Kubic A. Towards a formalization of emergence. *J Artif Life*. 2003;9(1):41-65. doi:10.1162/106454603321489518
77. Zeigler B, Mittal S, Traore M. MBSE with/out simulation: State of the art and way forward. *Article Sys*. 2018;6(4):18. doi:10.3390/systems6040040
78. Montgomery PR, Model-Based System Integration (MBSI) - Key attributes of MBSE from the system integrator's perspective. 2013:
79. Diallo SY, Lynch CJ, Gore R, Padilla JJ. Emergent behavior identification within an agent-based model of the Ballistic Missile defense system using statistical debugging. *J Def Model Simul-Appl Methodol Technol-JDMS*. 2015;13(3):275-289. doi:10.1177/1548512915621973
80. Enoiu E, Sundmark D, Causevic A, Pettersson P, A comparative study of manual and automated testing for industrial control software. ResearchGate; 2017:412-417.
81. Seth AK, Measuring emergence via nonlinear granger causality. 2008:545-553.
82. Tretmans J, Tangram: Model-based integration and testing of complex high tech systems. 2007.

AUTHOR BIOGRAPHIES

Rune André Haugen is an industrial-PhD candidate at the University of South-Eastern Norway (USN). He was in service with the Royal Norwegian Air Force (RNoAF) from 1997 to 2003, including graduation from the RNoAF Officer Candidate School in Stavern (1999) and the RNoAF Academy in Trondheim (2001). He holds both a BSc (2006) and a MSc (2013) in Systems Engineering from USN. He has worked as a design engineer at FMC Kongsberg Subsea from 2006 to 2008 (3D modeling), and as a system engineer at Kongsberg Defence and Aerospace since 2008 (system design and system test).

Nils-Olav Skeie got his MSc in Cybernetics from Norwegian University of Science and Technology (NTNU) in 1985. He worked with system development within the computer, aviation and maritime industry for more than 20 years before receiving a PhD within machine learning in 2008 in a cooperation between NTNU and the University of South-Eastern Norway (USN). In 2006 he went back to the academia and has been teaching BSc, MSc and PhD students in software engineering and system engineering. He continued to work as a part time system architect for the maritime industry from 2008 to 2015. He became a professor of industrial machine learning at USN in 2020.

Gerrit Muller, originally from the Netherlands, received his MSc in physics from the University of Amsterdam in 1979. He worked from 1980 until 1997 at Philips Medical Systems as a system architect, followed by 2 years at ASML as a manager of systems engineering, returning to Philips (Research) in 1999. Since 2003 he has worked as a senior research fellow at the Embedded Systems Institute in Eindhoven, focusing on developing system architecture methods and the education of new system architects, receiving his PhD in 2004. In January 2008, he became a full professor of systems engineering at the University of South-Eastern Norway. He continues to work as a senior research fellow at the Embedded Systems Institute in Eindhoven in a part-time position. All information (System Architecture articles, course material, curriculum vitae) can be found at: Gaudí systems architecting http://www.gaudisite.nl/

Elisabet Syverud is an Associate Professor in systems engineering at the University of South-Eastern Norway. She has 20 years of work experience from the industry, including 3 years at KDAs missile division as part of the Systems Engineering group. She received her MSc in Aerospace Engineering from the University of Kansas, US, and her PhD in Thermal Energy from the Norwegian University of Science and Technology in Trondheim, Norway. She started her industrial career in 1993 and has worked in multiple roles in the oil & gas and defense industries for almost 20 years. Since 2019, she is Associate Professor of Systems Engineering and Head of Department Science and Industry Systems at USN.

How to cite this article: Haugen RA, Skeie N-O, Muller G, Syverud E. Detecting emergence in engineered systems: A literature review and synthesis approach. *Systems Engineering*. 2023;26:463–481. https://doi.org/10.1002/sys.21660

APPENDIX A: LITERATURE SEARCH RESULTS

TABLE A1 Primary references found through database searches.

Index	Primary references	Citations	Publication channel	Affiliations
1	Axelsson[10]	0	INCOSE International Symposium 2022	Mälardalen University and RISE Research Institutes of Sweden
2	Beale and Tryfonas[45]	5	INCOSE International Symposium 2018	University of Bristol
3	Bondar, Hsu, Pfouga and Stjepandić[64]	64	Journal of Industrial Information Integration	PROSTEP California State University
4	Callister and Andersson[37]	2	INCOSE International Symposium 2016	Aker Solutions University of South-Eastern Norway
5	Chan, Son and Macal[73]	135	Proceedings of the 2010 Winter Simulation Conference	Rensselaer Polytechnic Institute The University of Arizona Argonne National Laboratory
6	Chan[74]	39	Proceedings of the 2011 Winter Simulation Conference (WSC)	Rensselaer Polytechnic Institute
7	Diallo, Lynch, Gore and Padilla[79]	7	Journal of Defense Modeling and Simulation-Applications Methodology Technology-Jdms	Old Dominion University VMASC of ODU
8	Donaldson[43]	18	Systems Engineering	Christopher Newport University
9	Freund, Al-Majeed and Millard[48]	0	16th International Conference on System of Systems Engineering, SoSE 2021	University of Lincoln
10	Freund, Al-Majeed and Millard[47]	0	16th International Conference on System of Systems Engineering, SoSE 2021	University of Lincoln
11	Guariniello, Khalid, Fang and Delaurentis[1]	9	Systems Engineering journal 2020	Purdue University University of Science and Technology, Wuhan, China
12	Harris and Narkevicius[55]	2	INCOSE International Symposium 2016	Rational LLC Jenius LLC
13	Hessami and Karcanias[6]	5	INCOSE International Symposium 2011	Vega Systems Systems and Control Research Centre
14	Holland[33]	27	Agent Directed Simulation Symposium, ADS 2007	Naval Surface Warfare Center, Dahlgren Division
15	Hsu and Butterfield[2]	20	INCOSE International Symposium 2007	The Boeing Company
16	Hsu, Axelband, Rouse, Madni and Sheard[5]	0	INCOSE International Symposium 2008	The Boeing Company RAND Corporation Tennenbaum Institute Intelligent Systems Technology Third Millennium Systems
17	Hsu, Axelband, Madni, Dagli and McKinney[4]	0	INCOSE International Symposium 2009	Queens University RAND Corporation Intelligent Systems Technology Missouri University Lockheed Martin
18	Hsu, Clymer, Garcia Jr. and Gonzalez[75]	4	INCOSE International Symposium 2009	Queens University California State University Southern California Edison
19	Hyun, Song, Shin, Baek and Bae[68]	1	Asia-Pacific Software Engineering Conference	Korea Advanced Institute of Science and Technology
20	Khan and Wang[49]	0	2016 IEEE International Conference on Electro Information Technology (EIT)	Bradley University

(Continues)

TABLE A1 (Continued)

Index	Primary references	Citations	Publication channel	Affiliations
21	Kjeldaas, Haugen and Syverud[8]	2	INCOSE International Symposium 2021	University of South-Eastern Norway
22	Kossiakoff, Flanigan, Seymour and Biemer[65]	33	Systems Engineering Principles and Practice	Johns Hopkins University
23	McConnell[12]	0	INCOSE International Symposium 2000	BAE SYSTEMS
24	McConnell[14]	5	INCOSE International Symposium 2001	BAE SYSTEMS
25	McConnell[13]	4	INCOSE International Symposium 2002	BAE SYSTEMS
26	McConnell[15]	0	INCOSE International Symposium 2012	Finmeccanica
27	Mittal, Diallo and Tolk[16]	23	Emergent Behavior in Complex Systems Engineering: A Modeling and Simulation Approach	MITRE Corporation Old Dominion University
28	Mnif and Muller-Schloer[39]	13	2006 IEEE Mountain Workshop on Adaptive and Learning Systems	University of Hannover
29	Montgomery[78]	9	Conference on Systems Engineering Research, CSER 2013	Naval Postgraduate School
30	Neace and Chipkevich[67]	4	IEEE National Aerospace and Electronics Conference, NAECON 2018	Johns Hopkins University
31	Osmundson, Huynh and Langford[3]	13	INCOSE International Symposium 2008	Naval Postgraduate School
32	Paunovski, Eleftherakis and Cowling[66]	10	Computing and Informatics	South-East European Research Centre (SEERC) City College University of Sheffield
33	Pickard and Beasley[46]	0	INCOSE International Symposium 2022	Rolls-Royce
34	Pourafzal and Fereidunian[72]	2	2020 6th Iranian Conference on Signal Processing and Intelligent Systems (ICSPIS)	Toosi University of Technology
35	Raman and Jeppu[57]	3	14th Annual IEEE International Systems Conference, SYSCON 2020	Honeywell Technology Solutions Lab
36	Raman and Jeppu[58]	1	15th Annual IEEE International Systems Conference, SysCon 2021	Honeywell Technology Solutions Lab
37	Raman and Murugesan[61]	0	INCOSE International Symposium 2022	Honeywell Technology Solutions Lab Honeywell Aerospace
38	Raz, Llinas, Mittu and Lawless[56]	8	22nd International Conference on Information Fusion, FUSION 2019	Purdue University University at Buffalo United States Naval Research Laboratory Paine College
39	Shin, Hyun, Shin, Song and Bae[69]	1	16th International Conference on System of Systems Engineering, SoSE 2021	Korea Advanced Institute of Science and Technology (KAIST)
40	Yang, Chen, Lu and Zhao[70]	13	5th International Conference on System of Systems Engineering, SoSE 2010	National University of Defense Technology
41	Zeigler, Mittal and Traore[77]	26	Systems	University of Arizona MITRE Corporation University of Bordeaux
42	Øvergaard and Muller[54]	1	INCOSE International Symposium 2013	University of South-Eastern Norway

TABLE A2 Secondary references found through references in the primary references.

Index	Secondary references	Citations	Publication channel	Affiliations
1	Adcock, Jackson, Fairley, Singer and Hybertson[44]	–	Systems Engineering Body of Knowledge	Stevens Institute of Technology
2	Bedau[29]	142	Nous journal	Reed College
3	Cummings[63]	3	PhD dissertation	Naval Postgraduate School
4	Dunn[50]	38	Process Improvements Using Data	DH Pace Company, Inc
5	Enoiu, Sundmark, Causevic and Pettersson[80]	15	IEEE International Conference on Software Testing, Verification and Validation, ICST 2017	Maelardalen University
6	Fromm[32]	80	arXiv	University of Kassel
7	Giammarco[36]	16	12th International Conference on System of Systems Engineering, SoSE 2017	Naval Postgraduate School
8	Haugen and Ghaderi[53]	0	SIMS EUROSIM 2021	University of South-Eastern Norway
9	Haugen and Mansouri[17]	2	INCOSE International Symposium 2020	University of South-Eastern Norway Stevens Institute of Technology
10	Holland[30]	1803	Basic Books	University of California
11	Johnson[41]	200	Reliability Engineering and System Safety journal	University of Glasgow
12	Kopetz[34]	408	Real-Time Systems 2011	Vienna University of Technology
13	Kopetz, Bondavalli, Brancati, Frömel, Höftberger and Iacob[40]	39	Cyber-Physical Systems of Systems: Foundations–A Conceptual Model and Some Derivations: The AMADEOS Legacy 2016	Vienna University of Technology University of Florence Resiltech SRL Thales Delft
14	Kubic[76]	126	Journal of Artificial Life	Silesian University
15	McLaughlin[38]	199	Emergence or Reduction?: Essays on the Prospects of Nonreductive Physicalism 1992	Rutgers
16	Mittal and Rainey[11]	54	SummerSim: summer simulation multi-conference	Dunip Technologies Integrity Systems and Solutions
17	Mogul[31]	154	Proceedings of the 1st ACM SIGOPS/EuroSys European Conference on Computer Systems	HP Laboratories Palo Alto
18	Montgomery[51]	16	Design and Analysis of Experiments, 8th ed.	Arizona State University
19	Murugesan and Raman[62]	0	ICONS: The Sixteenth International Conference on Systems 2021	Honeywell Aerospace Honeywell Technology Solutions Lab
20	Raman and Jeppu[59]	5	INCOSE International Symposium 2019	Honeywell Technology Solutions Lab
21	Raman, Gupta and Jeppu[60]	0	INCOSE International Symposium 2021	Honeywell Technology Solutions Lab
22	Ross and Aristotle[9]	–	Oxford: Clarendon Press	Cambridge University
23	Seth[81]	37	Artificial Life XI: Proceedings of the Eleventh International Conference on the Simulation and Synthesis of Living Systems 2008	University of Sussex
24	Singh, Lu, Kokar, Kogut and Martin[71]	13	Symposium on Modeling and Simulation of Complexity in Intelligent, Adaptive and Autonomous Systems 2017	North-eastern University Lockheed Martin
25	Snowden[7]	–	The Cynefin Co	Cognitive Edge

(Continues)

TABLE A2 (Continued)

Index	Secondary references	Citations	Publication channel	Affiliations
26	Szabo and Teo[28]	16	Proceedings of the 2012 Winter Simulation Conference	University of Adelaide National University of Singapore
27	Taguchi, Jugulum and Taguchi[52]	–	Computer-Based Robust Engineering : An Essential for DFSS	Aoyama Gakuin University Massachusetts Institute of Technology (MIT) ASI Consulting Group
28	Tretmans[82]	10	Tangram: Model-based integration and testing of complex high-tech systems	Embedded Systems Institute, Eindhoven
29	Wolf and Holvoet[35]	389	International Workshop on Engineering Self-Organizing Applications 2004	University of Leuven
30	Zeigler[42]	11	The Journal of Defense Modeling and Simulation	University of Arizona MITRE Corporation

APPENDIX B: LITERATURE SEARCH LOG

TABLE B1 Search log for literature search in relevant databases.

Date	Database	Search words combined with AND, OR, NOT	Limitations	Results	Comments/Notes
08.11.2021	Web of Science (WoS)	TS = (Emergen* AND Behavio* AND Complex* AND System* AND Engineering AND Model* AND Simulati*)		51 15 (title)	–
08.11.2021	Web of Science (WoS)	TS = (Emergen* AND Behavio* AND Complex* AND System* AND “Machine Learning”)		41 8 (title)	–
08.11.2021	Web of Science (WoS)	TS = (Emergen* AND Behavio* AND System* AND Engineering* AND Model* AND Simulati* AND (Identif* OR detect*)		19 3 (title)	2 duplicates (title)
08.11.2021	Web of Science (WoS)	KP = (Emergen* AND behavio* AND (Identif* OR detect*))		36 0 (title)	–
08.11.2021	Web of Science (WoS)	AK = (Emergen* AND behavio* AND (Identif* OR detect*))		8 4 (title)	–
	Web of Science			155 28 (title) 15 (abs) 4 (text)	2 duplicates within WoS 1 roadmap
09.11.2021	SCOPUS	TITLE-ABS-KEY (*emergen** AND *behavio?r** AND *complex** AND *system** *AND engineering* AND *model** AND *simulat**)		50 13 (title)	–
09.11.2021	SCOPUS	TITLE-ABS-KEY (*emergen** AND *behavio** AND *complex** AND *system** AND *“Machine Learning”*)		107 9 (title)	–
09.11.2021	SCOPUS	*TITLE-ABS-KEY* (emergen* *AND* behavio* *AND* system* *AND* engineering *AND* model* *AND* simulat* *AND* identif* OR detect*)		114 7 (title)	–

(Continues)

TABLE B1 (Continued)

Date	Database	Search words combined with AND, OR, NOT	Limitations	Results	Comments/Notes
	SCOPUS			271 21 (title) 17 (abs) 8 (text)	8 duplicates with WoS 3 roadmap
10.11.2021	IEEE	("All Metadata":Emergen*) AND ("All Metadata":Behavio*) AND ("All Metadata":Complex) AND ("All Metadata":System*) AND ("All Metadata":Engineering) AND ("All Metadata":Model*) AND ("All Metadata":Simulat*)		119 8 (title)	–
10.11.2021	IEEE	("All Metadata":Emergen*) AND ("All Metadata":Behavio*) AND ("All Metadata":Complex) AND ("All Metadata":System*) AND ("All Metadata":"Machine Learning")		28 5 (title)	1 duplicate (title)
10.11.2021	IEEE	("All Metadata":Emergen*) AND ("All Metadata":Behavio*) AND ("All Metadata":System*) AND ("All Metadata":Engineering) AN" ("All Metadata":Model*) AND ("All Metadata":Simulati*) AND ("All Metadata"Identifi* OR detect*))		72 6 (title)	2 duplicates (title)
05.10.2022	IEEE	("Document Title":Complexity) AND ("All Metadata":SoSE)	2014-2022	9 4 (title) 2 (abs)	
05.10.2022	IEEE	("Abstract":"Fault Knowledge Base") OR ("Abstract":"Fault Database") AND ("All Metadata":SoSE)	2014-2022	12 2 (title)	
	IEEE			240 22 (title) 20 (abs) 9 (text)	3 duplicates within IEEE 5 duplicates with Scopus
04.10.2022	Wiley	"Emergence" in Title and "Complexity" anywhere and "Systems Engineering" anywhere	Systems Engineering OR INCOSE International Symposium	44 19 (title) 15 (abs)	
05.10.2022	Wiley	Automatic* in Title and Test* in Title and Regression anywhere	Systems Engineering OR INCOSE International Symposium	1 1 (title)	
05.10.2022	Wiley	"System-of-Systems" in Title and Complexity in Keywords and "Emergent Behavior" anywhere	Systems Engineering	3 2 (title) 1 (abs)	
07.10.2022	Wiley	"Systems Engineering" anywhere and "Emergence" in Abstract and "Complicated System" in Abstract and "Complex System" in Abstract	INCOSE International Symposium	4 2 (title)	
11.10.2022	Wiley	"Systems+Engineering+Practice" in Title and "Emergence AND Complex System AND System Integration" anywhere	2020-2022 Books	16 5 (title) 1 (text)	

(Continues)

TABLE B1 (Continued)

Date	Database	Search words combined with AND, OR, NOT	Limitations	Results	Comments/Notes
11.10.2022	Wiley	“Technical+Debt” in Title and “System Integration” anywhere and “”Systems Engineering”” anywhere		4 1 (title)	
	Wiley			72 29 (title) 20 (abs) 20 (text)	1 duplicate with SCOPUS
05.10.2022	Science Direct	Title, abstract, keywords: Model Based System Engineering Title: Model Based System Integration		18 1 (title)	
	Science Direct			18 1 (title) 1 (abs) 1 (text)	

Note: We could preferably update the search words to do this screening process more automatically by using the NOT operator. Some records were duplicates in between the different search words we used in the same database, and some records were duplicates in between the different databases. A few of the records were put on a roadmap for later study because they were focusing more on the tools than the methods. For some records, we were not able to retrieve the full text documents.