pp 1545–1560. © Royal Aeronautical Society 2019
doi:10.1017/aer.2019.51

A holistic engineering approach to aeronautical product development

I. Staack
ingo.staack@liu.se
Department for Management and Engineering
Linköping University
Sweden

K. Amadori and C. Jouannet
Department for Management and Engineering
Saab AB/Saab Aeronautics
Linköping University
Sweden

ABSTRACT

Product development, especially in aerospace, has become more and more interconnected with its operational environment. In a constant changing world, the operational environment will be subjected to changes during the life cycle of the product. The operational environment will be affected by not only technical and non-technical perturbations, but also economical, managerial and regulatory decisions, thus requiring a more global product development approach. One way to try tackling such complex and intertwined problem advocates studying the envisioned product or system in the context of *system of systems* (SoS) engineering. SoSs are all around us, probably in any field of engineering, ranging from integrated transport systems, public infrastructure systems to modern homes equipped with sensors and smart appliances; from cities filling with autonomous vehicle to defence systems.

Since also aerospace systems are certainly affected, this work will present a holistic approach to aerospace product development that tries spanning from needs to technology assessment. The proposed approach will be presented and analysed and key enablers and future research directions will be highlighted from an interdisciplinary point of view. Consideration of the surrounding world will require to look beyond classical engineering disciplines.

Keywords: Morphological analysis; System of systems; System engineering; Matrix-Based design; Techniques for SoS Engineering; Standardisation; Rationalisation; Decision support; Interactive dashboard; Heterogeneous domains; Ontology; Semantic

Received 2 May 2018; revised 31 March 2019; accepted 16 May 2019.
A version of this paper was presented at the 31st ICAS Congress of the International Council of the Aeronautical Sciences in Belo Horizonte, Brazil in September 2018.

NOMENCLATURE

AI	artificial intelligence
CONOPS	concept of operations
DSL	domain-specific language
DSM	design structure matrix
IRMA	interactive reconfigurable matrix of alternatives
LoI	level of interest
MBSE	model-based systems engineering
MDDSM	multi-domain DSM
MDO	multidisciplinary (design) optimisation
NN	neural networks
QFD	quality functional deployment
RDF	resource description framework
SE	systems engineering
SoS(E)	system of systems (engineering)

1.0 INTRODUCTION

Product development in aerospace has been affected by very long lead times. Predicting the future is impossible, and forecasts become more uncertain the further into the future they need to stretch, thus leading to high levels of uncertainty regarding for instance available technologies or market assumptions. Especially within military development, a typical envisioned usage (*concept of operation* (CONOPS)) of a complex system will certainly change during its life cycle, due to changing, emerging or other unforeseen external factors that significantly influence the validity of the system. Traditional product development approaches based on an optimisation with respect to a fixed set of requirements fail to provide resilience in a constantly changing world. The problem becomes even worse when considering the long product life cycle that aerospace systems are designed for. Furthermore, since today's aerospace products are often part of a larger integrated system, a *system of systems* (SoS), it is important for the system manufacturer to be able to understand the relationships that lead from SoS needs, to required SoS capabilities, to requirements placed on single constituent systems. Customers may have performed detailed SoS analyses to produce a specification document for a constituent system to be developed. However, the product manufacturer needs to fully understand the customers' specifications and the underlying reasons. To engage in requirement discussions and negotiations, suggesting trading certain requirements while demonstrating that the overall needs will still be met, the manufacturer must to be able to carry out similar analyses to those carried out by the customer. Additionally, the manufacturer may wish to trade some requirements to achieve a better alignment of the future product to its own business strategy, to the overall product portfolio, to technology development plans and to the currently available and future-planned in-house competence. Also, the same product may be developed for different customers at the same time, imposing a more holistic view, since particular needs may diverge and simply producing a union of different requirement may lead to suboptimal solutions.

https://doi.org/10.1017/aer.2019.51 Published online by Cambridge University Press

1.1 SoS engineering

According to the definition offered by Maier[1], in this paper a SoS is assumed to possess five characteristic properties that sets it apart from conventional complex systems:

- Operational independence of the components
- Managerial independence of the components
- Geographic distribution of the components
- Emergent behaviour of the system
- Evolutionary development of the system

The importance of the last point listed above with respect to product development is also highlighted in the INCOSE SE handbook[2], which lists 31 product development processes for product life cycle engineering, which may be required concurrently in a huge SoS with its underlying systems in different life cycle stages and parallel system upgrades. Unlike classical (model-based) product development, huge efforts have to be invested to address the SoSs' evolutionary and emergent behaviour, which can occur at various levels[3]. Extensive forecasting and foresighting methods may be applied to analyse the *system of interest* (SoI) ramifications due to changes in the surrounding environment, external factors and other conditions (see Section 3.4). The used technology assessments should have the capability to identify disruptive technologies that may lead to disruptive events and emergent behaviours. One method for addressing the latter is *disruptive technology assessment games* (DTAG)[4], but more conventional approaches like matrix-based assessment methods (see Section 3.2) may also be applied for technology assessment[5].

In order to distinguish between different SoS-specific characteristics, Gideon et al.[6] proposed a taxonomy classifying every SoS by three type subsets only. Applying Gideon's taxonomy to large, complex aerospace SoSs, the following classification may apply: The SoSs are of *physical* domain type, most probably of a *dedicated* acquisition type and could be of any of the three operational types, *directed*, *collaborative* or *chaotic*. This work does not address a particular SoS within this classification, but rather tries to identify and define the different phases needed to approach the development of SoSs regardless of the type or the operational domain.

1.2 SoS research

Work performed by ASDL at Georgia Tech[7–9] has proposed methodologies to tackle SoS in the context of defined scenarios and requirements. SEAri at MIT[10,11] has chosen a different approach to the problem, focusing instead on epoch influences on development of complex systems and SoS (see Section 3.6). When addressing SoSs, expanding to a larger scope also implies that traditional engineering domains may not be sufficient. Stakeholders' needs may be dependent on socio-economical changes, and therefore a broader set of domains must be understood and integrated.

From literature reviews such as that presented by Axelsson[12], it can be noticed that *SoS engineering* (SoSE) is not yet fully defined as a scientific discipline, and therefore no clear and holistic handbook, guidelines or best practices addressing the whole design process exist. For this reason, this paper tries to offer a complete mapping of all perspectives within an overall SoS design process (as depicted in Fig. 1), including potential methods and tools that may support each phase. The goal is to outline a set of heuristics for SoS engineering and resilient design, but without proposing or developing deeper analytical methods.

https://doi.org/10.1017/aer.2019.51 Published online by Cambridge University Press

1.3 The SoS engineering paradigm shift

While conventional product development is primarily a technical-focused process within established domains, modern approaches like DARPA FANG(13) propose instead to tackle product development based on cyber-physical simulations and model integration by means of some kind of a *multidisciplinary design optimisation* (MDO) framework (e.g. AGILE(14)). These approaches still belong to the mechanical engineering domain, where huge progress has been made with respect not only to model implementation and modelling languages (like Modelica, Catia, Python, etc.), but also to available computational power and industry standards for model exchange and co-simulation such as the *functional mock-up interface* (FMI)(15). The primary concern of such solving frameworks is the early integration of physics-based models or methods of higher fidelity levels into the design process for design space exploration and optimisation. Generally, the foundation of such frameworks relies on a parametric geometry model that serves as the central node to which domain-specific models are connected as functional extensions(16).

In order to add higher fidelity and include non-mechanical engineering domains, the field of study has to be extended to an interdisciplinary *systems engineering* (SE) approach. This paradigm shift adds several new domains and concepts to the design process, the most important of which are addressed in Section 3. These extensions not only expand the design process upstream and downstream, but also introduce new domains and features to the design task such as business aspects, requirements and stakeholder needs handling, together with technology selection including technology maturation planning.

2.0 HOLISTIC PRODUCT DEVELOPMENT IN THE SOS CONTEXT

A holistic approach to product development in the context of SoS is proposed and illustrated in Fig. 1. The goal of this phase-based process decomposition is to identify the main areas of interest in order to tie needs, capabilities, and system requirements in the initial phases of product development. Five main *levels of interests* (LoI) have been identified, as follows:

- Needs and boundary conditions
- SoS capabilities
- SoS design space
- Constituent systems design space
- Sub-systems design space

The breakdown is recurrent and the main links between them are described in Fig. 1. The following section gives a brief overview of each phase and the connection to the adjacent levels:

Level of Interest 1 – Needs and Boundary Conditions

Within the product needs analysis, the needs related to the end-user needs are being analysed. The end-user needs and boundary conditions arising from a predicted environment are considered and analysed to determine the needs affecting the product. It is important to stress that such needs and boundary conditions are not intended to be limited to stakeholder requirements. Typical high-level frames of interest at this level are:

https://doi.org/10.1017/aer.2019.51 Published online by Cambridge University Press

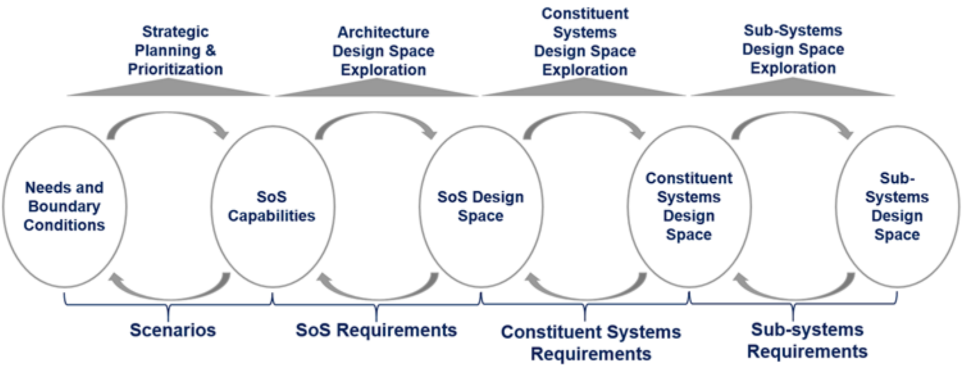
Strategic Planning & Prioritization
Architecture Design Space Exploration
Constituent Systems Design Space Exploration
Sub-Systems Design Space Exploration
Needs and Boundary Conditions
SoS Capabilities
SoS Design Space
Constituent Systems Design Space
Sub-Systems Design Space
Scenarios
SoS Requirements
Constituent Systems Requirements
Sub-systems Requirements

Figure 1. Overview of the proposed system of systems holistic design process.

- Geopolitics, doctrine, laws and regulations
- Business cases
- Customer needs
- Threats and technologies
- Time frame (history, now, future), needs and boundary conditions

These analyses can be related to a fixed period or to different time frames, meaning that all of those inputs will be characterised by different levels of uncertainties and may vary within the different time frames. From a holistic perspective, those initial conditions and boundaries have to be varied in order to understand the main required capabilities in response to changing needs. The output is a set of different scenario-representing needs. These scenarios should be agnostic to any solution to understand the main capabilities required by the SoS.

Level of Interest 2 – SoS Capabilities

The SoS capabilities are defined by scenario analyses. The underlying task is to figure out the impact of changes in the boundary conditions and the needs on the overall SoS capabilities. This analysis process leads to a balanced definition of the overall requirements on the SoS. Here, the capabilities design space is explored with the aim of understand it and providing decision support for strategical choices. The output will provide the main capabilities to be considered in the subsequent SoS design space exploration phase.

Level of Interest 3 – SoS Design Space

With help of the architecture design space exploration, the SoS capabilities are transformed into a SoS design space containing all valid solutions that achieve the desired capabilities. Out of this pool, possible SoS concepts – including type and number of the constituent systems, collaboration and tactical models – are generated, responding to the different identified capabilities. Each SoS concept is represented by one entry in the SoS design space. This design space is then down-selected by benchmark processes to a short list of designs, each made of a set of constituent systems. As an output, each constituent system will have a set of individual requirements.

https://doi.org/10.1017/aer.2019.51 Published online by Cambridge University Press

Level of Interest 4 – Constituent Systems Design Space

Based on the individual constituent systems requirements, the design space for each constituent system is generated. Conceptual design of each constituent system is then performed based on the requirements provided by the SoS design space analyses. This phase will validate the feasibility of each envisioned constituent system of the short-listed design space. It corresponds to the traditional product development process where one product or system is designed from a set of requirements.

Level of Interest 5 – Sub-Systems Design Space

Sub-systems are the systems that constituent systems are made of. The sub-system design task includes alternative architectures, system dimensioning and characterisation, and compatibility and integration into complete constituent systems. Typically, the sub-system analyses will consist of models for each discipline within a constituent system. The process can be interpreted as a whole (classic) conceptual design phase for each system, preferably implemented in a highly integrated *model-based system engineering* (MBSE) approach, enabling the analysis of a large number of different architectures and configurations.

2.1 Comparison to established product development processes

The proposed holistic SoS approach extends the usual design and product development processes to also include all boundary conditions (from geopolitics to regulations, from environment to time relations), as described in the LoI 1-3 in Fig. 1. Unlike established *design-to-cost* (DTC)(17) driven or *design-to-value* (DTV) driven approaches, the emphasis is more on a thorough system capabilities analysis (in LoI 2) that defines the SoS design space, and not the product setup. Taking into account the whole life cycle analysis of the system as defined by ISO 15288(18), the proposed approach goes far beyond classic *multidisciplinary design optimisation* (MDO) or CAD/CAE/CAM tools or frameworks usually applied for product development within industry. These tools are tailored for analysing and exploring constituent systems' and subsystems' design spaces (see LoI 4 and 5 in Fig. 1).

Another issue the reader should be aware of is that the lack of a universally accepted definition of SoS leads to confusion about whether a product can be seen as a complex system (that might contain several subsystems of different domains, e.g. a military aircraft) or a SoS. Applying the rather strict and distinct SoS definition proposed by Maier(1) (see Section 1.1), many as SoS denoted systems do not match this classification. Industrial examples for such pseudo-SoS systems include interface and (software) architecture concepts like the AUTOSAR architecture(19) for automotive applications.

3.0 KEY ENABLER FOR HOLISTIC DESIGN ENGINEERING

Each process phase described in Section 2 involves its own challenges if it is to be realised; some of them are more mature in methods and tools than others. A higher level of abstraction will be necessary to combine the different phases into one framework. This section presents a selection of different methods, research results and fundamental techniques that are identified by the authors as key enablers or available solutions in order to realize the envisioned process.

https://doi.org/10.1017/aer.2019.51 Published online by Cambridge University Press

3.1 Meta-modelling and common language

In order to connect the different domains of the design phases illustrated in Fig. 1, a common language and semantics is required. Using ontology to describe a complex system or a complete SoS may be the way forward(20). A complete ontology (description) of the system of interest might theoretically represent a complete information input for a SoS design process beside the requirements. While examples of ontology for aeronautical applications can be found in publications(21–23), the usefulness of this approach for complex system/SoS engineering has yet to be proven. In a similar way, the DARPA FANG(13) and DARPA AVM(24) projects focused on decreasing the product development time through component-based design and efficient cross-domain modelling. There was a great emphasis on the development of a model integration language, CyPhyML(25), and an semantic backplane OpenMeta(26,27). The selected tool for formal meta-modelling was FORMULA from Microsoft Research, a framework for formally specifying *domain-specific languages* (DSLs) and model transformations(28).

From a mathematical point of view category and sheaf theory(29) could be the foundation for an axiomatic description of the problem or the design space. This mathematical foundation seems promising and despite the fact that more applied research is needed to prove its usefulness, it has recently been acknowledge by DARPA as a cornerstone of the DARPA FUN design(30). Another approach to represent large and complex SoSs has been applied by military organisations through the usage of *system modelling language* (SysML); creating an enterprise architecture approach to capture the information about the business to identify the processes and resources required to deliver the vision expressed in the strategy. Different variants of these architecture frameworks, depending on their origin, are available(31). Prominent examples are DoDAF(32)/MODAF(33)/NAF(34) and IDEAS for military applications. The different architectures contain different viewpoints(32):

- Strategic viewpoint
- Operational viewpoint
- Service orientated viewpoint
- Systems viewpoint
- Acquisition viewpoint
- Technical viewpoint
- All viewpoints

These standards have the advantage of being based on a universal system modelling language, but have not yet been proven to be usable within product development as a main backbone for the execution of model-driven design processes (unlike in the software engineering domain with e.g. *executable UML* (xtUML) models). Combinations of such framework- and service-oriented architectures may enable the execution of SoS within its different viewpoints. Such a framework will serve as the link between viewpoints and models. The creation of domain-specific models, however, will still need to be performed in other frameworks/languages.

3.2 Matrix-based approaches

Matrix-based information arrangement is a common and natural choice of representing (any type of) relationship between different entities. Introduced in 1985 by Steward(35) for product (development) modelling, it is usually denoted as *design structure matrix* (DSM). In a certain arrangement, linking the customer needs to the system characteristics, it is called

https://doi.org/10.1017/aer.2019.51 Published online by Cambridge University Press

the *quality functional deployment* (QFD), also known as the house of quality. One implementation strategy of a user-in-the-loop matrix-based product development process is the *interactive reconfigurable matrix of alternatives* (IRMA) process (see e.g. Engler et al.[(7)]). While the mathematics/logic relations in these (usually 2$^{\text{-dim}}$) matrices are simple, the applied processes on the matrices – namely sorting and clustering – are not; each of these processes represents a local optimisation problem, fighting with the inherent problem of the sheer unfathomable number of combinations in already small-/medium-sized matrices[(16)]. The combinatoric growth is at least exponential with the power of two over the matrix size (number of matrix fields), but becomes significantly higher for clustering operations (e.g. with the target of sorting and detecting bus/integrator instances within a matrix[(36)]). Inherent problems of 2$^{\text{-dim}}$ matrices in the $n^{\text{-dim}}$ design space include the fragmentation of clusters and acausal relationships[1]. Due to the break-up into a forward and a backward part of interconnected entities/modules, the matrix-based representation becomes difficult to read; this effects not only large and complex systems but also low complexities as low as triple or tetrahedron cluster formations[(37)].

Some single-domain *DSM* drawbacks can be mitigated by adding more domains to the *DSM*, extending the usual square 2$^{\text{-dim}}$ (*NxN*) matrix into a composite 3$^{\text{-dim}}$ (*NxNxD*) matrix with *D* different domain matrices. However, due to the absence of a natural diagrammatic (2$^{\text{-dim}}$) representation of a multi-domain 3$^{\text{-dim}}$ structure, graphical solutions to represent *multidomain DSMs* (MDDSMs) have to be found. A possible decomposition of a 3$^{\text{-dim}}$ space into a 2$^{\text{-dim}}$ space can be achieved by cascading the data and presenting the higher dimension within the cells of the first and second dimensions. Abstraction can be achieved by the application of rating schemes e.g. those devised by Pimmler and Eppinger[(38)], and extended later by Helmer et al.[(36)].

A significant difference between intra-system and intra-SoS relations is that most systems' relationships within a SoS are communication channels for information exchange while physical system relationships often deal with the exchange of matter such as materials, fluids, energy, forces and heat. Consequentially, suitable modelling approaches (and tools) differ for both applications such as *UML* and *SysML* for the former and Modelica or Simulink for the latter.

3.3 Relational/graph-based

With the named disadvantages of matrix representations at hand, one solution to describe the system of interest is a graph network. With the help of 3$^{\text{-dim}}$ rendering, colour schemes, arrows and entities/cluster size, several domains can be represented in a human-understandable manner on a 2$^{\text{-dim}}$ screen provided that the network entities have been arranged (and if necessary clustered or sorted) with the help of suitable layout (positioning) algorithms such as Fruchterman-Reingold[(39)] or Hu[(40)]. Schaeffer[(41)] lists the different mathematical approaches for graph clustering that can be applied for product modelling.

The advent of huge social networks and the associated data mining and analysis needs triggered the development of various tools, relational database systems and data formats for graph structures (such as Gephi[(42)]). Defining a relational network and editing can be carried out without any knowledge of the residue data unlike in a hierarchical databases approach such as the classic product tree structure. Every relation in the relational database/network

[1] Mathematical, logical and physical relationships such as matter exchange are usually acausal/bidirectional.

https://doi.org/10.1017/aer.2019.51 Published online by Cambridge University Press

is a resource-trait/aspect-resource triplet that establishes the relationship between two entities. These relational entities are data triples similar to the *resource description framework* (RDF) triples used to model ontology within the semantic web approach, originally invented by Berners-Lee et al.[44] (see also Section 3.1).

3.4 Forecasting and foresighting methods

To define aerospace needs in future scenarios, forecasting or foresighting (see Section 3.4) must be performed. The goal of forecasting is to provide a prediction of highly probable future events, often based on an extrapolation of known facts. In contrast, foresighting does not aim to predicting the future but rather:

> *"... to explore the range of plausible futures that may emerge and to help identify assumptions and strategies that are robust in preparing for an uncertain future."[44]*

Several different forecasting and foresighting methods exist and have been summarised by Kindvall et al.[45], Cho and Daim[46]. The selective data collection process (typically executed by subject matter experts) will lead to recommendations for technologies and scenarios that have been identified as the most influential ones, see example from Silfverskiöld et al.[47]. One inherent drawback of these methods is the subjective judgement that may affect the results. One key to using the findings from such methods would be to transform these scenarios and technology recommendations into models that can be part of the framework described in Fig. 1. The application of foresighting within a framework for SoS engineering has been presented by Ross and Rhodes[48] and will be addressed further in Section 3.6.

3.5 Value-driven and robust design

Value-driven design aims to shift the focus from the requirements only to understand and analyse the value for the customer brought into the SoS by different parts of the design[49]. Underlying resectioning is required to tie customer needs to the added value created by the different solutions. Methods proposed by Isaksson and Bertoni[50–52] within aerospace applications show promising results and could be a valuable asset within the envisioned holistic product development process.

3.6 Epoch analysis

Traditional systems engineering tends to focus on meeting technical requirements. However, in a dynamic world, assumptions will probably change over time, affecting both technical and non-technical factors[49]. One method to address these changes over time is the epoch analysis proposed by Rhodes and Ross[10,48]. Beesemyer et al.[53] defines an epoch as:

> *"...a period of time, defined by a fixed set of context and needs, which impacts the ultimate success of a system. A long-lived system may face a large number of epochs over its lifetime."*

The work performed by the Rhodes SEAri group at MIT has shown the practicality of epoch analyses on various applications ranging from aerospace[53–57] to maritime[58]. Application has mainly been on large complex systems, with some extensions to SoS[11,59]. The authors of this paper feel confident that epoch analysis methods can be a key enabler for setting up the first LoI in the proposed holistic development approach.

https://doi.org/10.1017/aer.2019.51 Published online by Cambridge University Press

3.7 Data-driven design and tradespace exploration

Tradespace exploration is not only a way of assessing more design solutions than current methods allow for. It is also envisioned to be an interactive visual environment, enabling live *what-if* questioning to cover more criteria than are commonly applied in early conceptual design phases. The goal is to provide resilient system solutions in a changing context and a long-term perspective inherent to future aerospace SoSs. To perform such tradespace analyses, a data-driven approach is mandatory to enable an unremitting evaluation and analysis of alternatives. Data-driven methods rely on large computations with sensitivity analyses performed on all relevant variables. In contrast to current approaches, where requirements are considered as the primary input for product development, the aim of tradespace exploration is to generate the system requirements(60). Tradespace exploration techniques and diverse applications have been presented to a large extent by the MIT SEAri group(61–66). The U.S. Department of Defense (DoD) funded recently the Engineered Resilient Systems (ERS) project(67) to explore more efficient methods for military acquisition. As a result of the ongoing effort, the *DoD* also wants to leverage data-driven design.

3.8 Visual analytics and big data

The authors recognise the need to incorporate big data handling coupled with efficient interactive visualisation as a key capability. The different design spaces within each phase of the proposed process will lead to a very large set of data that needs to be managed and understood to support well-informed decision making. Georgia Tech has for a long time advocated using visual analytics as an assistive technology for decision support(68,69) to make large SoS design space explorations and uncertainty quantifications possible. Also within military applications, visual analytics and big data are being identified as key enablers for efficient acquisition of military products in the future (see the previously mentioned *ERS* project of the US *DoD*). The Swedish Defence Research Agency (FOI) recently published a comprehensive summary of the current research state of visual analytics methods(70).

3.9 Other domains

Most of the identified SoS enablers in this section originate from engineering domains. However, to realise the envisioned holistic development approach, additional domains have to be investigated and understood to benchmark their impact and capabilities concerning the design space exploration. Some key thoughts are presented here. They should not be seen as a definitive list but rather as the current status of the authors' knowledge.

Economic decision-making studies performed by the economist Thaler and co-workers(71–73) incorporates psychologically realistic assumptions, limited rationality, social preferences and lack of self-control of the stakeholders. These studies show that external factors have a large (non-rational) influence on decision making. Consequently, similar methods and assumption must be incorporate into the product development process, where customer preferences may certainly be influenced by similar factors.

The availability and recent progress of *artificial intelligence* (AI) can be an opportunity for decision support and large data analyses within the context of trade studies. It may also support better domain-specific understanding as well as helping to identify advantageous and disadvantageous emergent cross-domain coupling effects. Further understanding of current research (from the authors' point of view, all with an engineering background) is needed to incorporate the non-engineering disciplines such as geopolitical modelling and assessment

https://doi.org/10.1017/aer.2019.51 Published online by Cambridge University Press

into future implementations. Application examples that may largely benefit from ongoing machine learning and natural language research are meta-modelling and socio-economical domain representations.

4.0 CURRENT LIMITATIONS AND THE WAY FORWARD

This work describes a holistic approach to *system of systems engineering* (SoSE) in the context of aerospace. To solve the specific SoS needs and challenges during (SoS) product development, the authors of this work suggest a five-level process (explained in Section 2 and Fig. 1), largely expanding the traditional product development processes. Our research identifies various methods and techniques applicable to these five phases. An overview of the domains and application areas of these methods, described in detail in Section 3, is summarised in Table 1.

Table 1 illustrates common areas of application (such as aerospace, maritime products and doctrine descriptions) and system levels of application, and shows in which phase of product development the techniques are used. The table indicates the environment in which the methods are applied: academia, research institutes, industry and governmental organisations. Serving as a qualitative maturity indicator, the product development phase columns in the table are only filled in if the methods have been found to be used as state-of-the-art within the industry[2].

Paying heed to all named domains within one holistic SoS approach currently appears to be infeasible due to the overwhelming complexity and the different modelling approaches within each field. Another reason for this diversity, besides the broad variety of work scopes, is the lack of an established *holistic SoS* research and education field. Consequently, existing solutions are biased by the research groups' backgrounds such as mechanical engineering, computer science, social psychology, mathematics and so forth. With today's knowledge, it is not clear whether a distributed (master-master) framework of different domain experts will be the solution or a single master domain has to be found to take charge of the whole orchestra. Can a symphony lead by different domain conductors produce the desired outcome?

A central point in the implementation strategy has to be the decision between a *machines first* or a *humans first* approach[(74)]. How much of the design process can and has to be understood by the person involved? How can the output be actively influenced? How can the operator be integrated into the tool machinery? Can it be ensured that the tradespace reduction process (prior to the actual system development process) is correct and does not exclude any relevant areas of interest? And finally, which methods and processes are able to identify and foresee emergent behaviours of the SoS? A purely AI-like behaviour might be not acceptable due to sensitivity and traceability requirements for trade-off analyses. Relying on training-based AI methods – also denoted as big data mining – such as *neural networks* (NN) may not be the ideal solution for SoSs due to the lack of relevant training data, although it appears appealing to make use of an NN algorithm analysing the ontology description of the system of interest. The general absence of empirical data in SoS publications is also criticised by Axelsson[(12)]. It is possible that applying inspirational ideas from the software engineering domain will lead to widely accepted systematic methods and best practices within the SoS research community.

[2] The data collection is based on the examples found within the references. It is therefore non-exhaustive and might only serve an illustrative purpose.

https://doi.org/10.1017/aer.2019.51 Published online by Cambridge University Press

Table 1
Overview of the domain-specific methods.

- • Academia
- ◉ Research institute
- ○ Industry
- ● Governmental organization

		Aerospace	Pharmaceutics	Maritime	Land vehicle	Doctrine	Geo-politics and economics	Urbanization / power grids	SoS	System	Sub-System	Component	Initial Studies	Concept development	Preliminary development	Detail development	Maintenance
Meta-modelling and Common Language	Ontology	•◉○●	•◉			•◉ ●	•◉ ●		•◉ ●	•◉ ●	•◉ ●		▬	▬			
	Component-based Development			•◉	•◉○					•◉○	•◉○	•◉○	▬	▬	▬	▬	▬
	Domain-specific languages (DSL)																
	System modelling languages (UML/xtUML/SysML/etc.)	•◉○		•◉○	•◉○			•◉○	•◉○●	•◉○●	•◉○●	•◉○●		▬	▬	▬	
	Architecture frameworks (MODAF/DODAF/NAF/etc.)	•◉○●		•◉○●	•◉○●	●			●	●			▬	▬	▬	▬	
	System Dynamics	•◉		•◉	•◉	•◉	•◉	•◉	•◉	•◉							
Matrix-based Approaches	Design Structure Matrix (DSM)	•◉○●		•◉○●	•◉○●			•◉○●	•◉○●	•◉○●	•◉○●	•◉○●	▬	▬	▬	▬	▬
	Quality Functional Deployment (QFD)	●		•	●					•◉○●	•◉○●	•◉○●	▬	▬	▬	▬	▬
	Interactive Reconfigurable Matrix of Alternatives (IRMA)	•◉○		•◉○	•◉○			•◉○	•◉○	•◉○	•◉○	•◉○	▬	▬	▬	▬	▬
Relational / Graph-based	Resource Description Framework (RDF)	•◉		•◉	•◉			•◉	•◉	•◉	•◉	•◉					
	Graph Theory	•◉	•◉○	•◉	•◉	•◉	•◉	•◉	•◉	•◉	•◉		▬	▬	▬	▬	
Technology forcasting	DTAG/CTAG	◉○●		◉○●	◉○●				◉○●	◉○●	◉○●		▬				
	Data Mining	•◉○●	•◉○●	•◉○●	•◉○●	•◉○●	•◉○●	•◉○●	•◉○●	•◉○●	•◉○●	•◉○●	▬	▬			▬
Data-driven design	Value-driven Design	•◉○		•◉○	•◉○					•◉○	•◉○	•◉○	▬	▬	▬	▬	
	MDAO	•◉○	•◉○	•◉○	•◉○				•◉○●	•◉○●			▬	▬			▬
Decision support	Epoch Analysis	•◉ ●		•◉ ●	•◉ ●		•◉ ●		•◉ ●	•◉ ●							
	Tradespace Exploration	•◉○●	•◉○●	•◉○●	•◉○●		•◉○●		•◉○●	•◉○●			▬	▬			
	Visual Analytics	•◉○		•◉	•◉		●	•◉	•◉	•◉			▬	▬			
	Big Data	•◉○●	•◉○●	•◉○●	•◉○●		•◉○●	•◉○●	•◉○●	•◉○●	•◉○●	•◉○●	▬	▬			▬

Are there any further research fields SoS can be inspired by? Compared to SoS research classic fields of science are generally more mature and consequently their standards and best practices are widely accepted at an international level. Examples where complex collaborative work has largely contributed to scientific success include experimental physics, medicine, economic sociology (irrational behaviour) and climate change research. While the proposed process has not yet been realised, similar approaches are under development driven by the DoD (see DoD digital engineering[75]) with an estimated time frame of full scale deployment within the coming decade.

REFERENCES

1. Maier , M.W. Architecting principles for systems-of-systems, *Systems Engineering*, 1998, **1**, (4), pp 267–284.
2. Walden, D.D., Roedler, G.J., Forsberg, K.J., Hamelin, R.D. and Shortell, T.M. *INCOSE Systems Engineering Handbook: A Guide for System Life Cycle Processes and Activities*, 4th ed. International Council on Systems Engineering, John Wiley & Sons, 2015.
3. Holland , O.T. Taxonomy for the modeling and simulation of emergent behavior systems, *Proceedings of the 2007 Spring Simulation Multiconference*, *SpringSim '07*, Vol. **2**, Society for Computer Simulation International, San Diego, CA, USA, 2007, pp 28-35.
4. Andersson, K., Norsell, M., Svantesson, C.G. and Andersson, J. Förstudie angående DTAG-metodik, Tech. Rep., Swedish National Defence College (FMV), 2010, Swedish.
5. Amadori, K., Bäckström, E. and Christopher, J. Future technologies prioritization for aircraft conceptual design, *2018 AIAA Aerospace Sciences Meeting*, 1746. American Institute of Aeronautics and Astronautics, 2018.
6. Gideon, J., Dagli, C.H. and Miller, A.K. Taxonomy of systems-of-systems. *Conference on Systems Engineering Research*, Institute of Electrical and Electronics Engineers (IEEE), 2005, pp 356–363.
7. Engler, W., Biltgen, P.T. and Mavris, D.N. Concept selection using an interactive reconfigurable matrix of alternatives (IRMA), *45th AIAA Aerospace Sciences Meeting and Exhibit*, 1194. 2007.
8. Mavris, D.N. and Jimenez, H. An evolution of morphological analysis applications in systems engineering. *48th AIAA Aerospace Sciences Meeting Including the New Horizons Forum and Aerospace Exposition*, No. 972 in Aerospace Sciences Meetings. American Institute of Aeronautics and Astronautics, 2010.
9. Roberts, W., Griendling, K., Gray, A. and Mavris, D.N. Unmanned vehicle collaboration research environment for maritime search and rescue. *30th Congress of the International Council of the Aeronautical Sciences*. ICAS, 2016.
10. Rhodes, D.H., Ross, A.M. and Nightingale, D.J. Architecting the system of systems enterprise: Enabling constructs and methods from the field of engineering systems. *3rd Annual IEEE International Systems Conference*, Institute of Electrical and Electronics Engineers, 2009, pp 190–195.
11. Rhodes, D.H. and Ross, A.M. Shaping socio-technical system innovation strategies using a five aspects taxonomy. *7th European Systems Engineering Conference (EuSEC)*, International Council on Systems Engineering (INCOSE), 2010.
12. Axelsson, J. A systematic mapping of the research literature on system-of-systems engineering. *10th System of Systems Engineering Conference (SoSE)*, Institute of Electrical and Electronics Engineers (IEEE), 2015, pp 18–23.
13. Eremenko, P. Fang : fast, adaptable, next-generation ground vehicle, Tech. Rep., DARPA-BAA-12-15, Defense Advanced Research Projects Agency (DARPA), 2011.
14. DLR . AGILE: Aicraft 3rd generation MDO for innovative collaboration of heterogeneous teams of experts. Horizon 2020-EU.3.4, Project: 636202, 2018. [Online; accessed 2018-04-15]. URL https://www.agile-project.eu/.
15. Modelica Association . FMI-standard: Functional mock-up interface, Modelica Association, 2019. [Online; accessed 2019-01-14]. URL http://www.fmi-standard.org.

https://doi.org/10.1017/aer.2019.51 Published online by Cambridge University Press

16. STAACK, I. *Aircraft Systems Conceptual Design*. Linköping Studies in Science and Technology. Dissertation No. 1805, Department of Management and Engineering (IEI), Linköping University, Sweden, 2016.
17. MICHAELS, J. and WOOD, W. *Design to Cost*. New Dimensions in Engineering Series. Wiley, ISBN 9780471609001, 1989.
18. ISO 15288 . ISO/IEC/IEEE 15288:2015: Systems and software engineering – system life cycle processes. IEEE STD, 2015.
19. AUTOSAR . AUTOSAR methodology. Tech. Rep. V1.2.2 068, R3.2 Rev 1, Automotive Open System Architecture (AUTOSAR), 2006.
20. BENALI, H., BEN SAOUD, N.B. and BEN AHMED, M. in Information System Development: *Towards an Ontology of SoS Interoperability: Proposition of a SoS Interoperability Framework and a SoS Conceptual Interoperability Model*, chap. 7. Springer International Publishing, 2014.
21. AST, M., GLAS, M. and ROEHM, T. Creating an ontology for aircraft design: An experience report about development process and the resulting ontology. *Deutscher Luft- und Raumfahrtkongress*, 301256. Deutsche Gesellschaft für Luft-und Raumfahrt-Lilienthal-Oberth eV, 2013, pp 1-11.
22. REISS, M., MOAL, M., BARNARD, Y., RAMU, J.P. and FROGER, A. Using ontologies to conceptualize the aeronautic domain. *Proceedings of the International Conference on Human-Computer Interaction in Aeronautics*. Cépaduès-Editions, 2006, pp 56-63.
23. GLAS, M. *Ontology-based model integration for the conceptual design of aircraft*. Ph.D. thesis, Technical University Munich (TUM), Germany, 2013.
24. DE WECK , O.L. Fast adaptable next-generation ground vehicle challenge: Challenge analysis. DARPA # HR0011-13-C-0041 Phase 1 (FANG–1), Intelligent Action Inc., US, 2013.
25. SZTIPANOVITS, J., BAPTY, T., NEEMA, S., KOUTSOUKOS, X. and JACKSON, E. Design tool chain for cyber-physical systems: Lessons learned. *Proceedings of the 52nd Annual Design Automation Conference*, DAC '15. ACM, 2015, pp 81:1–81:6.
26. SIMKO, G., LEVENDOVSZKY, T., NEEMA, S., JACKSON, E., BAPTY, T., PORTER, J. and SZTIPANOVITS, J. Foundation for model integration: Semantic backplane. *32nd Computers and Information in Engineering Conference*, Vol. **2**, American Society of Mechanical Engineers (ASME), 2012, pp 1077-1086.
27. SIMKO, G., LINDECKER, D., LEVENDOVSZKY, T., NEEMA, S. and SZTIPANOVITS, J. Specification of cyber-physical components with formal semantics – integration and composition. *Model-Driven Engineering Languages and Systems*. Springer, Berlin, Heidelberg, 2013, pp 471–487.
28. JACKSON, E., LEVENDOVSZKY, T., and BALASUBRAMANIAN, D. Reasoning about Metamodeling with Formal Specifications and Automatic Proofs. *International Conference on Model Driven Engineering Languages and Systems (MODELS 2011)*. Association for Computing Machinery (ACM)/IEEE, 2011.
29. KASHIWARA, M. and SCHAPIRA, P. *Categories and Sheaves*. Grundlehren der Mathematischen Wissenschaften. ISBN 978-3-540-27949-5, Springer, Berlin, Heidelberg, 2006.
30. DARPA. Fundamental design (FUN DESIGN). Tech. rep., Disruption opportunity special notice DARPA-SN-17-71, Defense Sciences Office (DSO), DARPA, 2017.
31. KLEIN, J. and VAN VLIET, H. A systematic review of system-of-systems architecture research. *Proceedings of the 9th International ACM Sigsoft Conference on Quality of Software Architectures*, QoSA '13. ACM, New York, NY, US, 2013, pp 13-22.
32. DEPARTMENT OF DEFENSE . DoDAF V2.0 DoD architecture framework volume II: Architectural data and models. Version 2.02 Change 1, U.S. Department of Defense, 2015.
33. MINISTRY OF DEFENCE . MOD architecture framework guidance. Ministry of Defence, Military Equipment, logistics and technology, U.K., 2019. [Online; accessed 2019-03-12]. https://www.gov.uk/guidance/mod-architecture-framework.
34. NATO . NATO architecture framework. Version 4, NAFv4, Architecture Capability Team Consultation, Command & Control Board, North Atlantic Treaty Organization (NATO), 2018.
35. STEWARD , D.V. The design structure system: A method for managing the design of complex systems. *IEEE Transactions on Engineering Management*, 1981 **28**, (3), pp 71–74. Institution of Electrical Engineers (IEEE).
36. HELMER, R., YASSINE, A.A. and MEIER, C . Systematic module and interface definition using component design structure matrix. *Journal of Engineering Design*, 2010, **21**, (6), pp 647–675 .
37. YU, T., GOLDBERG, D.E., SASTRY, K., LIMA, C.F. and PELIKAN, M. Dependency structure matrix, genetic algorithms, and effective recombination. *Evolutionary Computation*, Vol. **17**. Massachusetts Institute of Technology, 2009, pp 595–626.

https://doi.org/10.1017/aer.2019.51 Published online by Cambridge University Press

38. Pimmler, T.U. and Eppinger, S.D. Integration analysis of product decompositions. *ASME Conference on Design Theory and Methodology*. American Society of Mechanical Engineers, 1994, pp 343–351.
39. Fruchterman, T.M. and Reingold, E.M. Graph drawing by force-directed placement. *Software – Practice and Experience*, 1991, **21**, (11), pp 1129–1164.
40. Hu, Y. Efficient and high quality force-directed graph drawing. *The Mathematica Journal,* 2006, **10**, (1), pp 37–71, Wolfram Media Inc.
41. Schaeffer, S.E. Graph clustering. *Computer Science Review*, 2007, **1**, (1), pp 27–64, Elsevier.
42. Bastian, M., Heymann, S. and Jacomy, M. Gephi: An open source software for exploring and manipulating networks. *International AAAI Conference on Weblogs and Social Media*. Association for the Advancement of Artificial Intelligence, 2009.
43. Berners-Lee, T., Hendler, J. and Lassila, O. The semantic web: A new form of web content that is meaningful to computers will unleash a revolution of new possibilities. *Scientific American*, 2002, Special Online Issue, **2**, pp 24–30.
44. Kuosa, T. *Towards strategic intelligence: foresight, intelligence, and policy-making*. Dynamic Futures, Vantaa, ISBN 978-952-68169-0-6, 2014.
45. Kindvall, G., Lindberg, A., Trane, C. and Westman, J. Exploring future technology development. Tech. rep., FOI-R-4200-SE, Swedish Defence Research Agency (FOI), ISSN 1650-1942, 2017.
46. Cho, Y. and Daim, T. *"Technology Forecasting Methods". Research and Technology Management in the Electricity Industry: Methods, Tools and Case Studies*. Springer, 2013, pp 67–112.
47. Silfverskiöld, S., Liwång, H., Hult, G., Sivertun, Å., Bull, P., Sigholm, J., Lundmark, M., von Gerber, C., Andersson, K. and Sturesson, P. Technology forecast 2017 – military utility of future technologies. Tech. rep., Report from Seminars at the Swedish Defence University's (SEDU) Military-Technology Division, 2017.
48. Ross, A.M. and Rhodes, D.H. Using natural value-centric time scales for conceptualizing system timelines through epoch-era analysis. *INCOSE International Symposium 2008*. International Council on Systems Engineering, 2008, pp 1–15.
49. Collopy, P.D. and Hollingsworth, P.M. Value-driven design. *Journal of Aircraft, American Institute of Aeronautics and Astronautics (AIAA)*, 2011, **48**, (3), pp 749–759.
50. Bertoni, M., Bertoni, A. and Isaksson, O. Evoke: A value-driven concept selection method for early system design. *Journal of Systems Science and Systems Engineering*, 2018, **27**, (1), pp 46–77, Springer, Berlin, Germany, Heidelberg.
51. Isaksson, O., Kossmann, M., Bertoni, M., Eres, H., Monceaux, A., Bertoni, A., Wiseall, S. and Zhang, X. Value-driven design: a methodology to link expectations to technical requirements in the extended enterprise. *23rd Annual International Symposium of the International Council on Systems Engineering*, Vol. **1**, International Council on Systems Engineering (INCOSE), 2013, pp 171–187.
52. Bertoni, M., Bertoni, A., Isaksson, O., Amnell, H. and Johansson, C. Value-oriented concept selection in aero-engine sub-systems design: the evoke approach. *23rd Annual International Symposium of the International Council on Systems Engineering*, Vol. **2**, International Council on Systems Engineering (INCOSE); 2013, pp 977–991.
53. Beesemyer, C., Ross, A.M. and Rhodes, D.H. Case studies of historical epoch shifts: Impacts on space systems and their responses. *AIAA Space 2012 Conference*. American Institute of Aeronautics and Astronautics, 2012, pp 1–13.
54. Rader, A.A., Ross, A.M. and Fitzgerald, M.E. Multi-epoch analysis of a satellite constellation to identify value robust deployment across uncertain futures. *American Institute of Aeronautics and Astronautics*. American Institute of Aeronautics and Astronautics, 2014, pp 1–16.
55. Pina, A.L. *Applying Epoch-Era Analysis for Homeowner Selection of Distributed Generation Power Systems*. Master's thesis, Massachusetts Institute of Technology, S.B. Aerospace Engineering, U.S., 2014.
56. Fitzgerald, M.E. and Ross, A.M. Mitigating contextual uncertainties with valuable changeability analysis in the multi-epoch domain. *2012 IEEE International Systems Conference*. Institute of Electrical and Electronics Engineers, 2012.
57. Fitzgerald, M.E. and Ross, A.M. Sustaining lifecycle value: Valuable changeability analysis with era simulation. *2012 IEEE International Systems Conference*. Institute of Electrical and Electronics Engineers, 2012.

58. GASPAR, H.M. and ERIKSTAD, S.O. Handling temporal complexity in the design of non-transport ships using epoch-era analysis. *International Journal Maritime Engineering*, 2012, **154**. Transactions RINA.
59. PARKER, V.D., ROSS, A.M. and RHODES, D.H. Program and portfolio affordability tradeoffs under uncertainty using epochera analysis. *INCOSE International Symposium*, Vol. **26**, no. 1, International Council on Systems Engineering, 2016, pp 2391–2406.
60. RHODES, D.H. and ROSS, A.M. Esd 411: Concept design and tradespace exploration. Course material, 2014. [Online; accessed 2018-03-27]. http://seari.mit.edu/documents/presentations/ESD411Oct14_Ross Rhodes_MIT.pdf.
61. ROSS, A.M., MCMANUS, H.L., RHODES, D.H. and HASTINGS, D.E. Revisiting the tradespace exploration paradigm: Structuring the exploration process. *AIAA 2010 Space Conference*. American Institute of Aeronautics and Astronautics, 2010, pp 1–14.
62. VARIOUS. Example mate projects, 2008. SEAri Working Paper Series, WP-2008-5-2, Massachusetts Institute of Technology. [Online; accessed 2017-11-13]. http://seari.mit.edu/documents/working_papers/SEAri_WP-2008-5-2.pdf.
63. ROSS, A.M. and HASTINGS, D.E. The tradespace exploration paradigm. *INCOSE International Symposium 2005*. International Council on Systems Engineering, 2005, pp 1–13.
64. ROSS, A.M., HASTINGS, D.E., WARMKESSEL, J.M. and DILLER, N.P. Multi-attribute tradespace exploration as front end for effective space system design. *Journal of Spacecraft and Rockets*, 2004, **41**, (1), pp 20–28.
65. SPAULDING, T. Mateing: Exploring the wedding tradespace. SEAri Working Paper Series, WP-2002-1-1, Massachusetts Institute of Technology, 2002. [Online; accessed 2017-11-13]. http://seari.mit.edu/documents/working_papers/ SEAri_WP-2002-1-1.pdf.
66. ROSS, A.M. *Multi-Attribute Tradespace Exploration with Concurrent Design as a Value-centric Framework for Space System Architecture and Design*. Master's thesis, Massachusetts Institute of Technology, Department of Aeronautics and Astronautics, 2003.
67. HOLLAND, J.P. Engineering for resilience. *AIAA SciTech Conference*. American Institute of Aeronautics and Astronautics, U.S., 2016.
68. MAVRIS, D.N., BALCHANOS, M., SUNG, W. and PINON, O.J. A data mining and visual analytics perspective on sustainability-oriented infrastructure planning. *Data Mining and Big Data*. Springer International, 2016, pp 330–341.
69. MAVRIS, D.N., PINON, O.J. and FULLMER, D.J. Systems design and modeling: A visual analytics approach. *Proceedings of the 27th International Congress of the Aeronautical Sciences (ICAS)*, 2010.
70. JANDEL, M., BIVALL, P., HAMMAR, P., JOHANSSON, R., KAMRANI, F. and QUAS, M.J. Visual analytics: Perspectives on the field of interactive visualization. Tech. rep., FOI, FOI-R-4200-SE, Stockholm, Sweden, 2016.
71. MULLAINATHAN, S. and THALER, R.H. Behavioral economics. NBER Working Papers 7948, National Bureau of Economic Research, Inc, 2000.
72. BONDT, W.F.M.D. and THALER, R.H. Financial decision-making in markets and firms: A behavioral perspective. NBER Working Papers 4777, National Bureau of Economic Research, U.S., 1994.
73. BARBERIS, N. and THALER, R. A survey of behavioral finance. NBER Working Papers 9222, National Bureau of Economic Research, U.S., 2002.
74. INCOSE . First INCOSE workshop on the evolution of human-systems integration. Tech. rep., Human-System Integration Working Group, Florida Institute of Technology, U.S., 2016.
75. DEPARTMENT OF DEFENSE . Digital engineering initiatives by the DoD. Office of the Deputy Assistant Secretary of Defense (ODASD), 2019. [Online; accessed 2019-03-17]. https://www.acq.osd.mil/se/initiatives/init_de.html.

https://doi.org/10.1017/aer.2019.51 Published online by Cambridge University Press