24th INTERNATIONAL DEPENDENCY AND STRUCTURE MODELING CONFERENCE, DSM 2022
EINDHOVEN, THE NETHERLANDS, 11 – 13 October, 2022

Using DSMs in functionally driven explorative design experiments – an automation approach

Massimo Panarotto[1], Timos Kipouros[2], Arindam Brahma[1], Ola Isaksson[1], Oskar Strandh Tholin[1], John Clarkson[2]

[1] Chalmers University of Technology, Gothenburg, Sweden
[2] University of Cambridge, Cambridge, United Kingdom

Abstract:

Product architectures are often designed as evolutions or modifications of existing product platforms by adding new functionalities. However, there is still a limited ability to simultaneously modify the functions and the physical elements of a system (and the corresponding links between them). These relationships, which are usually static, limit the ability to discover new architectures, as well as to analyse architectures by simulation. In this paper, we demonstrate how DSMs can be connected to function-means models to enable 1) the simultaneous modification of functions, solutions and links of a product's architecture and 2) the ability to export multiple DSMs from function models to use in simulations of non-functional properties of a product's architecture (e.g., supply chain resilience) in digital design experiments. The initially identified multiple instances of architectures can be exported and communicated with other simulation and analysis tools, thereby enriching existing models with additional and more detailed information.

Keywords: Design Structure Matrix, Function Modelling, Change Propagation, Design Support

1 Introduction

A green and sustainable future of air transport has led manufacturers to look for radically new solutions. In Europe alone, many such demonstration programs are already in their advanced stages (Clean Sky, 2020) yet, even more disruptive technologies are expected to emerge to achieve the sustainability goals. Manufacturers are investing in new engine concepts which play a major role in determining the sustainability impact of an aircraft. For instance, radically new engineering approaches are being explored in projects related to Open Rotor, Hybrid-Electric and Full Electric aircraft engines (Zaporozhets et al., 2020). Such approaches, however, have brought new sets of challenges. Despite the benefits in terms of, for example, CO2 reduction, these new configurations are much debated in terms of their architectural challenges. Such alternative engine concepts, therefore, will require new architectures and arrangements where multidisciplinary studies will need to be conducted on a wider range of variations in the concepts. Furthermore, sustainability being a multidimensional problem will require life cycle behaviour, natural resource utilisation and social impact to be included in any such studies.

For new engine architectures, the challenges for conducting these multidisciplinary studies are twofold. Firstly, these new architectures often establish new linkages with other parts of the system. For example, Open Rotor architectures have shown to present integration challenges both within the engines and to the aircrafts, stemming from nearly twice the diameter compared to current turbofans with uncontained rotor blades (SAFRAN, 2019). Stepping into an aircraft engine, the capacity to adapt to e-system design and its operative conditions sets the focus on reliability and resilient design solutions. Secondly, even if a promising technology is chosen during design, the ability to optimize and balance it with other design objectives is not readily available (Isaksson et al., 2021). The consequence is a risk that the final design becomes highly sensitive to change and variation in loads and modifications at later phases. To develop a product solution that is less sensitive to evolving and changing conditions (loads, design modifications, material changes), there is a need to consider the product's sensitivity to such changes during the design phases. Identification of failure modes not only from a performance perspective but also from a life cycle and sustainability perspective drives the need to understand and represent non-physical phenomena too. At present, design conditions are often assumed to be nominal and deterministic, and the ability to conduct probabilistic design studies are limited to later phases of development (Shergadwala et al., 2018). Further, the ability to quantitatively assess a design range of conceptual solutions, for instance, through Digital Experiments requires a high degree of automation to generate and analyse the many concepts in parallel.

Design Structure Matrix (DSM) based methods have been used effectively for several years to capture the relationships between elements (Browning, 2015) and to capture the degree of "invasiveness" when a new technology is introduced in a new system (Suh et al., 2010). Further, DSMs have also been used where it is difficult to obtain detailed data to quantify the non-functional properties of a design, such as risk or development process efficiency, especially in a preliminary design stage (Raudberget et al., 2014). Similarly, information extracted from DSMs have been used to run probabilistic methods such as the Change Propagation Method (Clarkson et al., 2004), which deals with predicting the susceptibility of components or subsystems to changes through multiple routes and the likelihood of propagation occurring at each such step. An approach towards the problem of generating radically new architectures as discussed previously could include running the aforementioned methods "in batch" for many different architectures and systems. this, however, would require new automation capabilities to be used at the DSM level.

At the same time, it is likely that new functionality and associated technologies will need to be integrated into new engine/aircraft arrangements, e.g., for cooling, shielding, controlled leakage and so forth. While DSMs are powerful representations of the connections between the physical elements of a system, they often do not contain information on the underlying design intent and rationale of a component or sub-system i.e., its underlying function (Müller et al., 2020). Often, this information is captured in other types of representations, such as function-means (F-M) models (Johannesson & Claesson,.2005). This static connection between F-M and DSM representations limits the ability to discover new architectures, as well as to analyse architectures by simulation. This is particularly relevant for products that are often designed as evolutions or modifications of existing product platforms by adding new functionalities, following a product platforms approach (Muffatto & Roveda, 2000).

These relationships, which are usually static, limit the ability to discover new architectures, as well as to analyse existing and new architectures by simulation. In this paper, we demonstrate how DSMs can be connected to function-means models to enable 1) the simultaneous modification of functions, solutions and links of a product's architecture and 2) the ability to export multiple DSMs from function models to use in simulations of non-functional properties of a product's architecture. Examples of such simulations could include supply chain resilience in digital design experiments. The benefits of the presented approach include the utilisation of historic data and DSM models that pre-exist in literature (and in previous projects within an organisation) and also to support a simulation-led architecture discovery methodology. In other words, the initially identified multiple instances of architectures can be exported and communicated with other simulation and analysis tools, thereby enriching existing models with additional and more detailed information.

2 Background

According to Ulrich & Eppinger (2011), a product architecture is *"the scheme by which the functional elements of a product are arranged into physical chunks and by which the chunks interact".* Depending on the ways this scheme is set up, the architecture can be classified into different types, for example, integral, bus or modular (Ulrich, 1995). This paper focuses on the alteration of an architecture independently of its type. Based on the definition by Ulrich & Eppinger (2011), this paper argues that changing any aspects of the arrangement or the interaction is essentially altering the architecture.

The design of a product architecture distinguishes two processes: architecture definition and architecture analysis (Ulrich, 1995). To support the definition of an architecture, function modelling has widely been adopted in industry, and its benefits have been discussed in the research literature (Eisenbart & Kleinsmann, 2017). Popular techniques for functional modelling are the Function Block Diagramming (FBD) and the Function Analysis System Technique (FAST). However, a limitation of these approaches is that they do not allow for the representation of design solutions and functions in the same model. Furthermore, these techniques do not allow the representation of architectural innovations where the functions and the solutions remain the same, but where the linkages are changed. For these reasons novel approaches of conducting function modelling have been proposed. This paper focuses on the application of the Enhanced Function-Means (EF-M) modelling method (Schachinger & Johannesson, 2000; Müller et al., 2019), which has been demonstrated in product platform modelling and design (e.g., Raudberget et al., 2014). The method is described in the next session.

2.1 The Enhanced Function-Means Modeling method

The EF-M model (Schachinger & Johannesson, 2000; Müller et al., 2019) enables the representation of both functions, solutions and interactions in the same model. The EF-M is a graphical representation of the different items (e.g., components or features) and their governing criteria (functions or constraints) that constitute a complex product. Both items and criteria are modelled as objects. Different relations are used to model how different objects are linked to each other. The objects used are:

- Functional requirements (FR): contain information about the functions that the system, shall provide – i.e. what the system and its subsystems should do.
- Design Solutions (DS): are the engineering solutions – or 'functional features' – that provide the functionality required by the Frs. The DSs are also defined as the parameters that describe the bandwidth of the physical characteristics of the system.
- Constraints (C): that limit the available solution space in any way. i.e. what are the solutions that can be allowed. Constraints are also defined with parameters that describe the bandwidth within which they can be applied.

The EF-M modelling approach organizes the relations between FRs, DSs and Cs in the same hierarchical model. EF-M follows a representation where only one DS solves one FR with cardinality 1:1 (a design principle referred to as the independence axiom (Suh, 1990)). This DS is in turn decomposed into two or more FRs. Each FR has one corresponding

DS attached to each of them. For the case of alternative design solutions, an FR can be realised by only one DS at a time, (Figure 2). As such, the E-FM can represent a family of alternative solutions. EF-M connects these objects according to six different types of relations:

- an FR is_solved_by (isb) a DP;
- a DS is_constrained_by (icb) a C;
- a DS requires_functions (rf) FRs on the next lower hierarchical level;
- a C is_partly_met_by (ipmb) DSs on the next lower hierarchical level;
- parallel solution DSs interacts_with (iw) each other;
- the fulfilment of an FR is_infuenced_by (iib) the choice of a parallel solution (DS).

The basic structure of an EF-M model, i.e. the objects and relations constituting the tree, is shown in Figure 1.

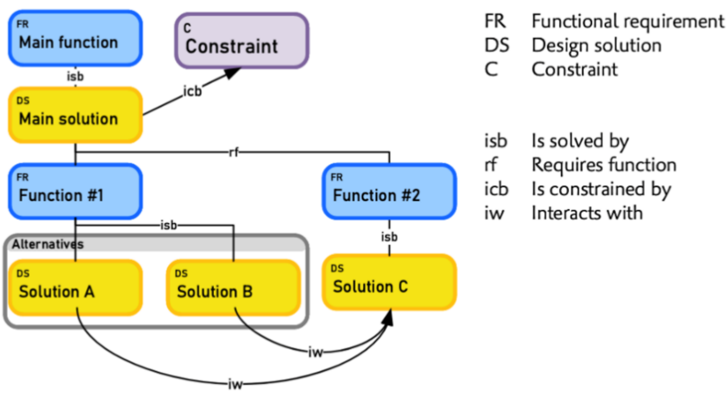

Figure 1. Basic structure of an Enhanced Functions-Means (EF-M) model. Adapted from Levandowski et al. (2014) and Muller (2020).

It is vital to note that this is a generic step – allowing alternative architectures to be defined and evaluated from genuinely early phases, where quite limited information is available about the forthcoming system. The generic nature also applies in situations where more and richer information about an architecture exists.

The analysis of the defined architectures often focuses on analysing the degree of coupling or modularity (Hölttä-Otto et al., 2012). This is because at this stage no detailed model of the architecture is available. Therefore, analysis methods based on Design Structure Matrices (DSMs) are helpful and effective at this stage, such as the Change Propagation Method (CPM: Clarkson et al., 2004). In this context, the EF-M method also supports the extraction of a DSM from the function model, which can be used for architecture analysis.

One limitation of function models and DSMs, independently from the techniques used, is that they do not allow the assessment of product dimensions (e.g., weight, drag) that would require representations with a higher degree of maturity (e.g., Computer Aided Design, CAD) from which simulations (e.g., from Finite Element Analysis, FEM) can be run. Recent research (Isaksson et al., 2021) has focused on connecting function models with geometry models, extending current Design Automation (DA) and Knowledge-Based Engineering (KBE) techniques. The next section focuses on reviewing such techniques.

3 Overall Methodology

The proposed methodology is shown in Figure 2, and leverages automation approaches applied at both functional as well as at the DSM level. To implement the methodology, the semi-automatic connection between two different enabling tools have been focused upon: Enhanced-Function Means Modeler and Cambridge Advanced Modeler (CAM).

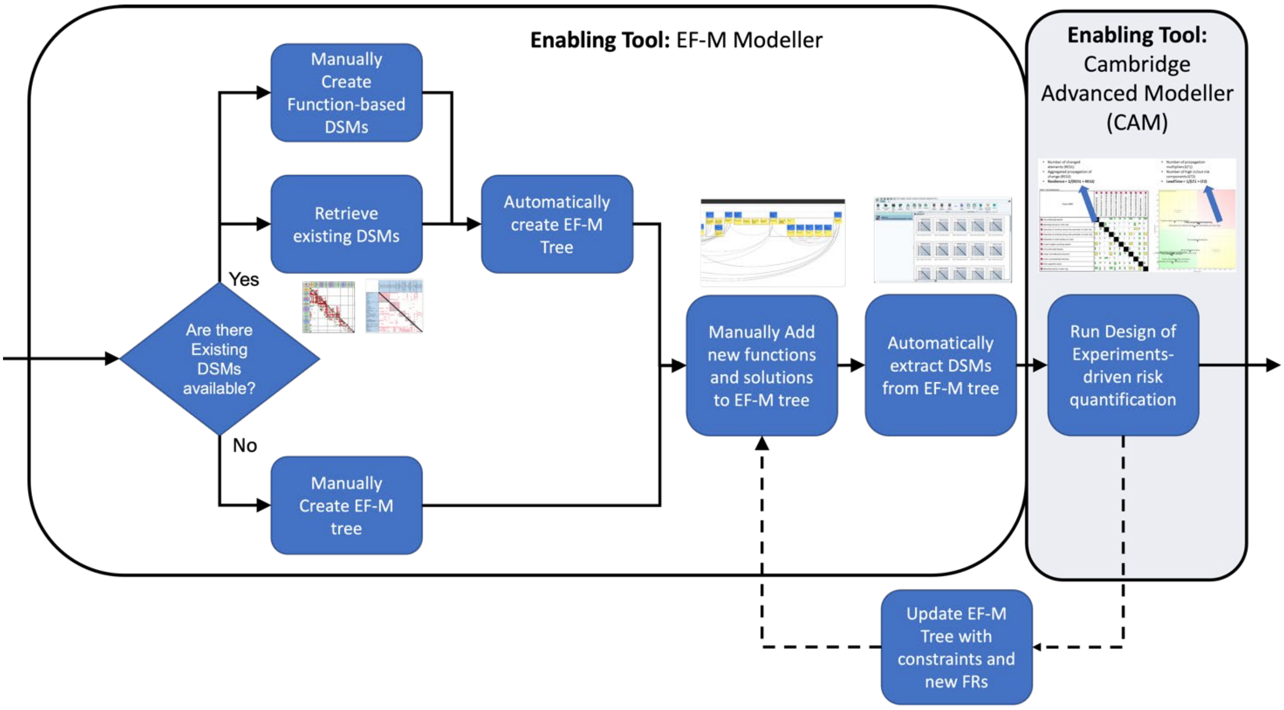

Figure 2. The overall methodology for functionally driven explorative design experiments of DSMs.

The approach starts by considering two situations:

1. There are DSM models that exist in the literature or in previous projects within an organisation. This is typical in many platform-based organizations that evolve their product through functional enhancements. In this case, it is possible to create an EF-M tree from these DSMs, through the addition of function-based information (called function-based DSMs).
2. There are no pre-existing DSMs. This is typical for a completely new system. In this case, the EF-M tree is built by capturing and considering the new desirable new functionalities, technologies and new architectures.

This modelling activity automatically generates a series of Design Structure Matrices as (DSMs) which represent all conceptual combinations of design concepts. These analyses effectively perform a quantified risk assessment through CPM (Change Propagation Method) using CAM. In the approach, it is intended to use the DSM/CPM analysis to inform back the EF-M model, through the addition of new constraints or update the current constraints. For example, the CPM analysis may reveal which components are influential and change propagators, and others to be the most susceptible receivers of change. This information can be used to add "constraints" to the EF-M model, so that designers can think of alternative ways of designing the component, so that the resulting system is less susceptible to change.

To illustrate the application of the proposed methodology, an industrial case is presented. A turbine rear structure (TRS) (see Figure 3) is an aerospace component located in the rear end of a jet engine.

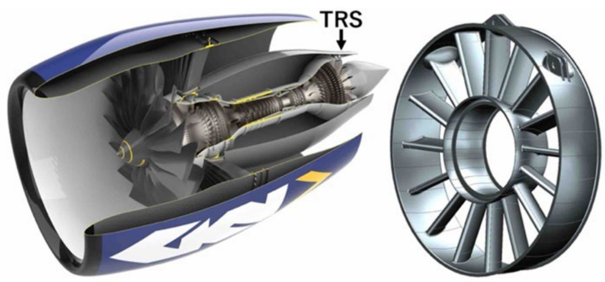

Figure 3. Basic structure of an Enhanced Functions-Means (EF-M) model. Adapted from Levandowski et al. (2014) and Muller (2020).

The TRS has a range of functional criteria from various fields of engineering. They must be able to withstand significant thermal and structural loads. Additionally, to optimize fuel efficiency, they must be as light and aerodynamic as possible. These functional criteria must be balanced to obtain an optimal design.

3.1 Manually create EF-M trees – in the case DSMs are not already available

Previous studies on architectural modelling (e.g., Jiao et al., 2007) have focused on modular products where there is a clear separation between the physical components. Also, in modular products, it is rather intuitive to associate each component with its underlying function. The TRS represents instead the case of a highly integrated product, meaning that all required functions are satisfied by a single, monolithic component (Raja et al. 2019). Therefore, EF-M modelling has been used to separate the elements of the system in order to identify the elements of this single monolithic component which contribute to satisfying the product's functions. Figure 4 shows how the EF-M technique has been used to represent functions, solutions and interactions between them.

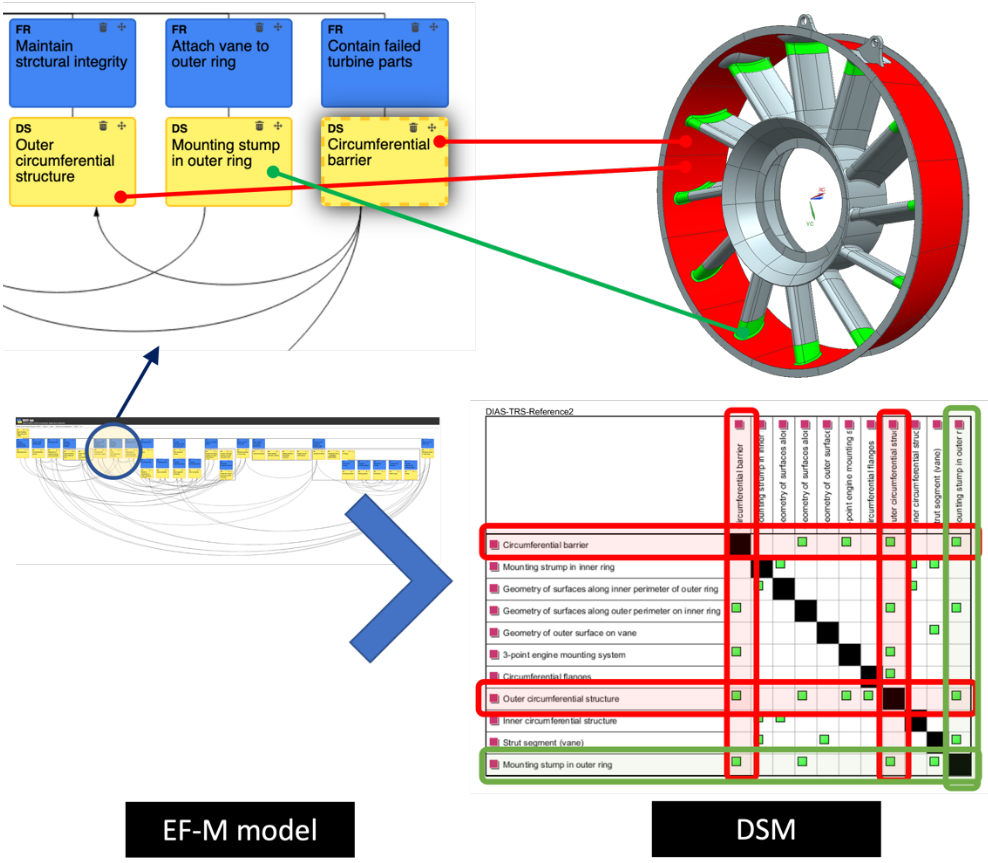

Figure 4. EF-M modelling of TRS and screenshot of the tree view in the tool, FR = functional requirement, DS = design solution.

For example, the upper left part of Figure 4 shows how the outer structure of the TRS fulfils two functions: 'Contain failed turbine parts' (solved by using the outer structure as a barrier) and 'Maintain structural Integrity' (solved with the actual outer structure). These two solutions present many interactions with other parts of the TRS (shown as red arcs in Figure 4). For example, a change in the diameter of the outer structure implies a change in the vane length (the grey parts in the TRS CAD model in Figure 4). However, the vane is not directly attached to the outer structure. This 'attach' function is solved in this TRS using 'mounting stumps' (the green parts in the TRS CAD model in Figure 4) in order to facilitate the vane attachment through welding. the EF-M model can support the representation of secondary effects between design solutions. For example, changing the number of vanes and the lean angles will change the number and the geometry of the mounting stumps, which in turn impact the performance of the outer circumferential structure (and its ability to act as a 'barrier' to contain failed parts). At the same time, these changes impact the welding time and welding performances.

The lower part of Figure 4 shows a screenshot of the EF-M modeller where these functions and solutions, through a web editor. The web application has been created by using a Python language to connect a database (using a Django Web Framework, https://www.djangoproject.com/) to a Web interface developed using the HTML coding language.

3.2 Automatically create EF-M Tree from existing DSMs

In the case existing DSMs are available from previous products (or from literature), the EF-M modeller allows to import a DSM, which is to be used as a reference system to create an EF-M tree. This is done by uploading three different Excel spreadsheets (in a single file), that must be named exactly as follows:

- *'FR-FR'*: This sheet contains a DSM-like structure that sets the function structure (i.e., parent and child function tree)

- *'FR-DS'*: This is a diagonal matrix that links each DS to the corresponding FR. The FRs in the rows are parents to the DSs in the columns. This matrix can be created by reverse engineering an existing DSM (e.g. interviewing engineers about the functions behind each component of the DSM).
- *'DS-DS'*: This sheet contains the existing DSM.

The three sheets containing the respective DSMs must not contain other data in the cells. Images and colouring of the cells (and other cosmetic formatting) are allowed on these sheets since they are ignored during the import. The DSMs can be placed anywhere on the sheet (i.e., does not have to be along the edges of the sheet). The connections in the DSMs can be indicated using whatever symbol or symbols the user wants. The program looks for non-empty cells when looking for connections. Therefore, it is important for the user to ensure that the empty cells really are empty. The project is named after the top-level DS.

3.3 Add new functions and solutions to EF-M trees

The EF-M tree of a TRS was modelled the EF-M modeller, and new functions and design solutions were implemented. This implementation is visualized in the lower part of Figure 5.

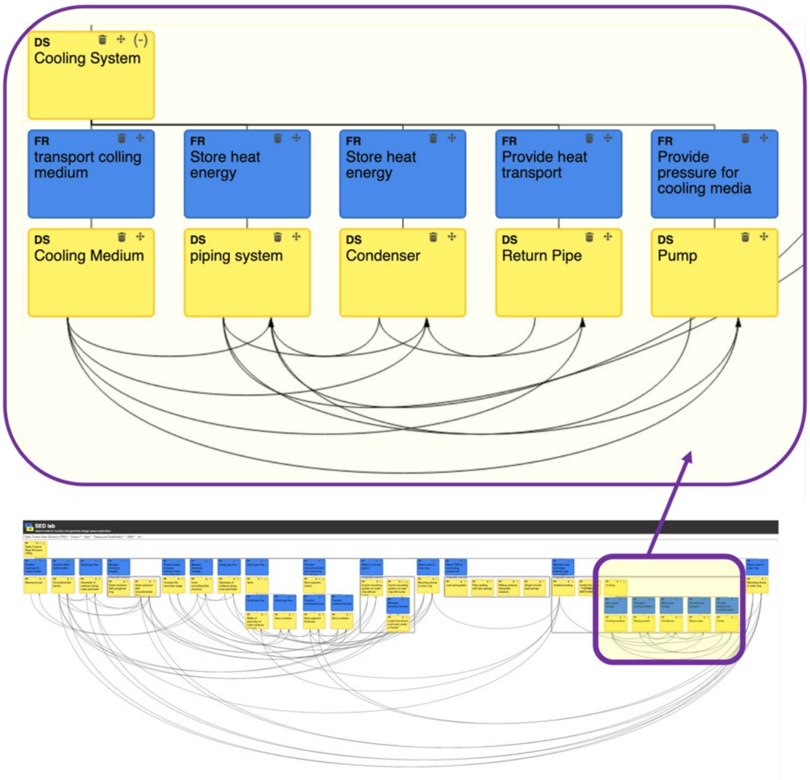

Figure 5. In the lower part, new functions and solutions are implemented in the EF-M modeller. In the upper part, a section of the "cooling system" DS is shown.

The tool supports three different ways to modify a product architecture:

1. Finding new design solutions that fulfil the same function (Ölvander et al., 2009): for example, a mechanical steering system fulfils the function of allowing the vehicle to follow a desired course. An electro-mechanical steering can be introduced to fulfil the same function, although arguably at a higher level of precision.
2. Introducing a new function which leads to a new technical solution (Gilain et al., 2019): For example, the Electronic Stability Control (ESC) system in a car allows the introduction of a new function - detecting a loss of steering control by the driver and automatically applying the brakes to support the steering of the vehicle.
3. Keeping the same technical solutions but changing the interactions and links between them: This way is referred to as 'architectural innovation' (Henderson & Clark, 1990). One example of an aerospace component is 'heat shields fairings'. Normally, a vane in a component fulfils both a guiding flow function and a carrying load function. The innovation in a fairing is to separate the links between these two functions, which leads to a solution for guiding flow (vane) and the carrying load function (strut).

In this particular case study, to the original TRS Architecture were added:

- 1 new DS as an alternative for the FR of *'maintain structural integrity'* is shown in Figure 4.

- 1 new FR *'Minimize the bending moment in the mounting system'* with a new DS associated
- 1 new FR to *'Attach the TRS to connecting component'* with 4 alternative DS
- 1 new FR to '*Minimise heat exchange between gas and material'* with 3 alternative DS.

This last FR is of interest for the context of this paper, as it can highlight how this modelling activity can allow the inclusions of DSs of multi-technological nature. Being a static component, a TRS has been historically made of mechanical solutions leveraging the aerodynamic and structural properties of material and geometries. For this new FR instead, new multi-technological solutions are added.

One of these solutions is shown in the upper part of Figure 5. It is a "cooling system" made out of a cooling medium, a piping system, a condenser, a pump and a return pipe. Compared to traditional solutions, this DS leverages mechanical elements combined with hydraulic solutions. These are shown in the upper part of Figure 5 as red links for the mechanical dependencies, and as blue elements for hydraulic (or fluidic connections).

This modelling activity performed on the EF-M tree enables to increase the level of innovation introduced in the system, ad solutions of different nature can be introduced and compared.

3.4 Automatically batch export DSMs from EF-M Tree and batch import DSMs in CAM

The EF-M tree modelling activity of the TRS generated 48 different concepts, which are the combinatorial sum of the different FR and DS added (i.e., 1 "reference" TRS × 2 DS for the structural integrity function × 2 DS for the bending moment × 4 DS for the function of attaching to connecting component × 3 DS for the minimizing heat exchange function).

The generated DSMs can be automatically extracted from the EF-M modeller either one-by-one, or in batch. At the same time, the generated DSMs can be imported in CAM, from one file per DSM, as shown in Figure 6. The analyses that CAM offer, for instance, CPM, can then be run on the workbooks containing the DSMs of the concepts.

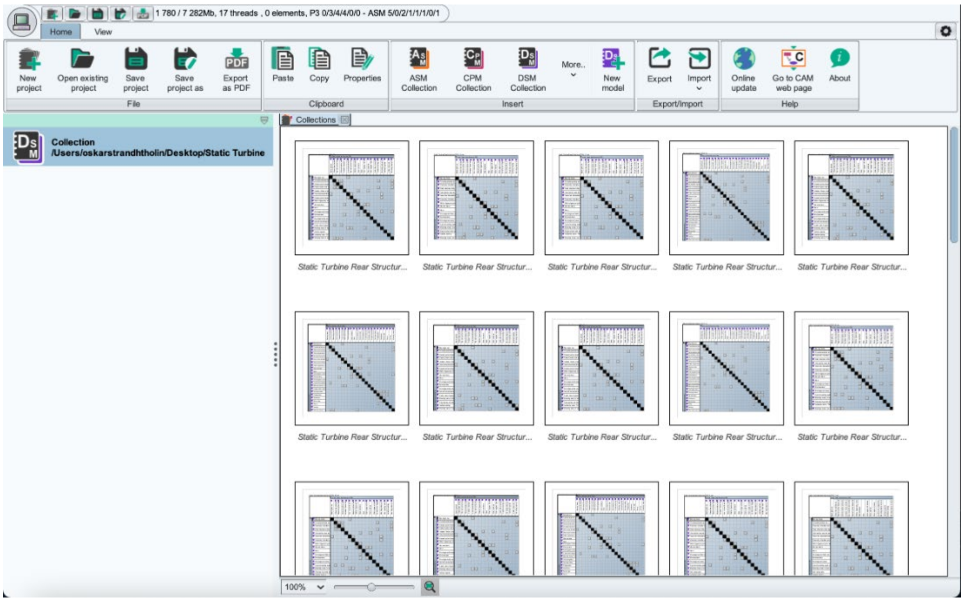

Figure 6. The user interface of CAM, with some of the workbooks containing the 48 exported DSMs, which have all been imported in batch.

CAM developed a dedicated functionality that allowed multiple .csv files to be imported at the same time. This allows users to run CPM analysis of multiple architectural alternatives at the same time.

3.5 Run Design of Experiments-driven risk quantification in CAM

As an example of a previously conducted small design study, eight alternative concepts were represented as DSM's, where the dependencies also carried likelihood and impact properties on each connection. Each concept, exported from E-FM to DSM was imported in CAM for CPM analysis. The Change Propagation Method in CAM (Clarkson et al., 2004) is a DSM-based numerical approach for predicting and analyzing how changes are likely to propagate through a system. The combined risk values are computed (Risk = Likelihood*Impact) and analyzed in the In/Out risk portfolio (Keller et al., 2009). This allows differentiating between change propagation absorbers and multipliers. Figure 7 shows the sequence for change propagation based on CPM.

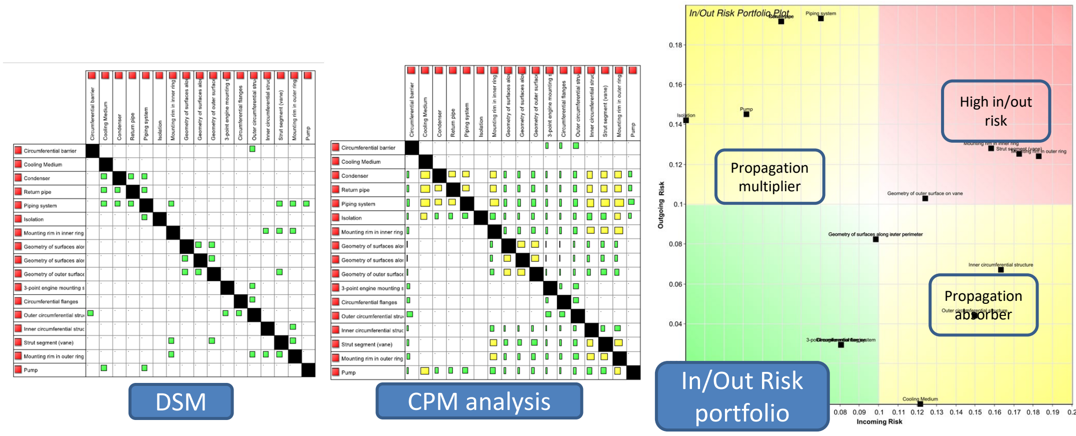

Figure 7. DSM from E FM run in Change Propagation and visualized in In/Out Risk portfolio

The designer can thus display the risk profile for multiple concepts and benefit from the automated export of DSMs and execution of propagation analysis. In the case above, the “piping system” was found to be the most influential initiator whereas the “mounting rim in outer ring” was the most susceptible receiver.

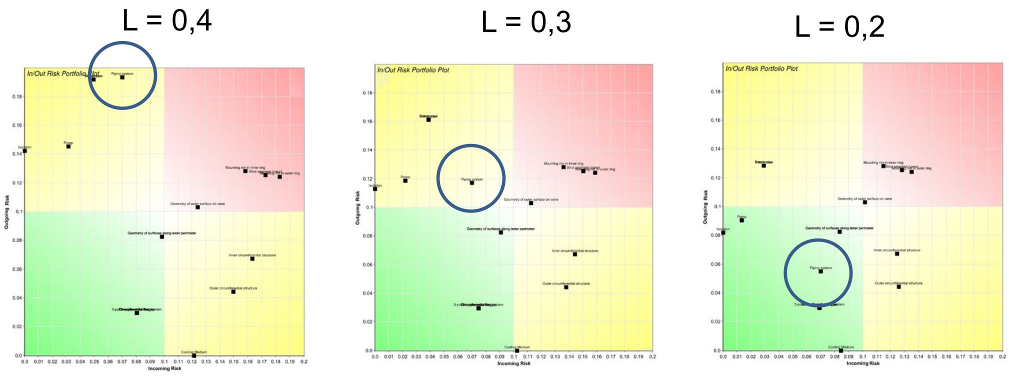

Figure 8. DSM from E FM run in Change Propagation and visualized in In/Out Risk portfolio

The designers were able to first understand the criticality of components using CPM, whereafter the effect of design solutions that reduce likelihood can be assessed. Evidently, introducing damping designs on the piping system could be a good strategy to improve robustness and resilience of the product. Figure shows the impact on the overall behaviour of the selected architecture when the likelihood of change of the piping system is reduced, keeping the impact of the change the same. Although all other components of the architecture maintain the same likelihood and impact values, it is noticeable that most propagation multipliers have improved behaviour.

3.6 Update EF-M Tree with Constraints and New FRs coming from the DSM/CPM analysis

Using CPM, the criticality of components in terms of risk of change - such as the piping system - has been highlighted. The DSM/CPM analysis results are used to inform back the EF-M model and add new constraints or update the current constraints. As reported in the background section, constraints are properties that in any way constrain the available solution space. i.e., what solutions can be allowed.

Since the CPM has highlighted how the piping system is the component that most propagates change, this can mean that the design of the piping system needs to be prioritized since the early stages, and carefully optimized. In other words, this component needs to be designed and “frozen” rather early in the process, as its change during the design process can imply changes to other components of the system.

Engineers can then ask what is the cause of changing the piping system (e.g., the fluid losses, which are directly connected to the diameter of the pipes). Therefore, a constraint on the pipe's diameter can be captured on the EF-M tree (Figure 9). This information can enable designers to understand which components need to be “frozen” early in the design stages, and also which components can be instead subject to modifications further down in the process. For example, the analysis in CPM has highlighted how the “mounting rim” is less susceptible to change (Figure 7). For engineers, this means that the design of this component can be “frozen” late in the design phases, as its change it is not going to affect other parts of the system. In the EF-M tree, this could result in a relaxation of the constraints connected to the mounting rim, in a process similar to that applied by Borgue at al. (2019).

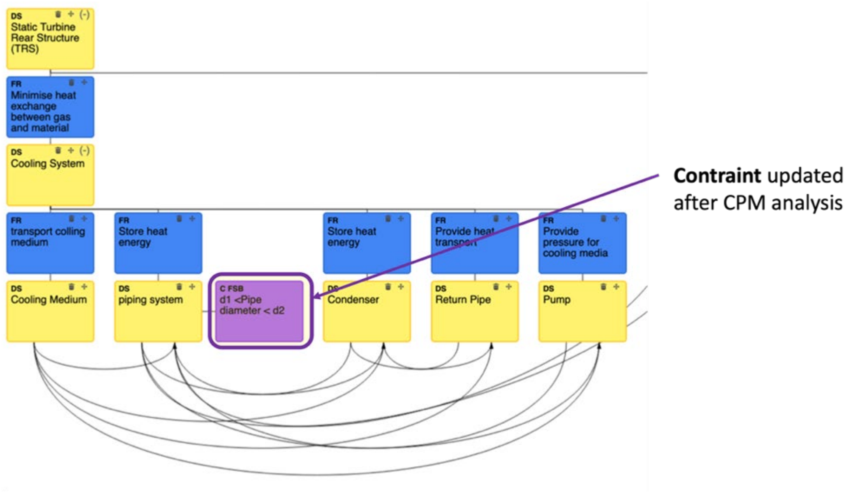

Figure 9. Update of EF-M constraints after running the CPM analysis.

Discussion and conclusion

In this paper, an automation approach to connect DSMs to function-means models has been presented. Automatically importing and exporting DSMs for different functionally defined concepts can be useful to enable the configuration of digital experiments in an automated fashion.

Automated and parametric design studies are already well established for later design phases, where design embodiment (e.g., a parametric CAD model) is available (Amadori et al., 2012). This enables the exploitation of design automation for the pre-embodiment phases (Pourtalebi & Horváth, 2016) as well.

Additionally, automatically exporting DSMs "in batch" allows quantified assessment of risk using tools such as CPM, complimenting traditional simulation techniques (e.g., to assess aerodynamics and mechanical performances) with the assessment of non-functional and non-geometrical properties of a design, such as susceptivity to change.

The ability of CPM to be used to inform the EF-M back, enables designers to resolve and consider "constraints" during the design activity, fostering continuous improvement and faster design iterations. The ability to capture feasibility and manufacturability constraints from physical testing activities has been explored in previous research (Borgue et al., 2019). This paper extends previous work by including "constraints" related to the design process itself, such as change propagation risks and supply chain resilience.

There are some aspects of the proposed methodology that deserve attention and future work. For example, the connection between the EF-M modeller and CAM is still semi-automatic. The DSMs need to be exported in batch as a zip file containing the DSMs as csv files, which are subsequently imported in CAM in batch through the same zip file. These connections will be made completely automatic in the future. From a methodological standpoint, other investigations are ongoing, which are looking into ways to include the ability to configure the digital experiments directly from the EF-M tree, for example by configuring a DoE with the likelihood and impact values to be communicated to CAM to run the CPM algorithm.

Conclusion

The value of a product is typically determined by how well the expected functions can be met. Further a product is typically comprised of sub-systems and components, the behaviour of which can be expressed by their interdependencies in DSMs. The presented work demonstrates how DSMs and EF-M models can be connected, which thereby enables designers to conduct various analyses on them. Such analyses include using change propagation assessment for risk and robustness analyses. Further, updates on the EF-M model can be carried out to evolve functional features and/or to design and therefore refine sub-solutions and their corresponding interrelations. It was also demonstrated that an automated export-import capability enables many design alternatives to be explored. It is therefore suggested that the demonstrated capability offers significantly improved design capacity, by an improved integration of functional design (using EF-M) and discrete analyses (using DSM). The design case of a structural jet engine component was used to show how design insights were gained and how they were used to improve the design accordingly.

Acknowledgments
The project leading to this paper has received funding from the Clean Sky 2 Joint Undertaking under the European Union's Horizon 2020 research and innovation programme. Grant agreement No 887174.

References

Browning, T. R. (2015). Design structure matrix extensions and innovations: a survey and new opportunities. IEEE Transactions on engineering management, 63(1), 27-52.
Amadori, K., Tarkian, M., Ölvander, J., & Krus, P. (2012). Flexible and robust CAD models for design automation. Advanced Engineering Informatics, 26(2), 180-195.
Borgue, O., Müller, J., Leicht, A., Panarotto, M., & Isaksson, O. (2019). Constraint replacement-based design for additive manufacturing of satellite components: ensuring design manufacturability through tailored test artefacts. Aerospace, 6(11), 124.
Clarkson, P. J., Simons, C., & Eckert, C. (2004). Predicting change propagation in complex design. J. Mech. Des., 126(5), 788-797.
Clean sky (2020), "Sustainable and Green Engines (SAGE)", . https://www.cleansky.eu/sustainable-and- green-engines-sage
Eisenbart, B., & Kleinsmann, M. (2017). Implementing shared function modelling in practice: experiences in six companies developing mechatronic products and PSS. Journal of Engineering Design, 28(10-12), 765-798.
Gilain, A., Le Masson, P., & Weil, B. (2019, July). The hidden feat behind development cost escalation-how engineering design enables functional expansion in the aerospace industry. In Proceedings of the Design Society: International Conference on Engineering Design (Vol. 1, No. 1, pp. 3011-3020). Cambridge University Press.
Pourtalebi, S., & Horváth, I. (2016). Towards a methodology of system manifestation features-based pre-embodiment design. Journal of Engineering Design, 27(4-6), 232-268.
Henderson, R. M., & Clark, K. B. (1990). Architectural innovation: The reconfiguration of existing product technologies and the failure of established firms. Administrative science quarterly, 9-30.
Hölttä-Otto, K., Chiriac, N. A., Lysy, D., & Suk Suh, E. (2012). Comparative analysis of coupling modularity metrics. Journal of Engineering Design, 23(10-11), 790-806.
Isaksson, O., Kipouros, T., Martinsson, J., Panarotto, M., Kressin, J., Andersson, P., & Clarkson, J. P. (2021). Multi-domain design assessment for aerospace components including weld accessibility. *Proceedings of the Design Society*, *1*, 2217-2226.
Jiao, J.R., Simpson, T.W. and Siddique, Z. (2007), "Product family design and platform-based product development: a state-of-the-art review", Journal of Intelligent Manufacturing, Vol. 18 No. 1, pp. 5–29.
Johannesson, H., & Claesson, A. (2005). Systematic product platform design: a combined function-means and parametric modeling approach. Journal of Engineering Design, 16(1), 25-43.
Keller, R., Eckert, C. and Clarkson, P.J. (2009). Using an engineering change methodology to support conceptual design. Journal of Engineering Design, 20(6), 571–587.
Levandowski, C., Raudberget, D. S., & Johannesson, H. (2014). Set-Based Concurrent Engineering for Early Phases in Platform Development. In ISPE CE (pp. 564-576).
Muffatto, M., & Roveda, M. (2000). Developing product platforms:: analysis of the development process. Technovation, 20(11), 617-630.
Muller, J. R. (2020). Does Form Follow Function?. Connecting Function Modelling and Geoemtry Modelling for Design Space Exploration, Chalmers University of Technology, Gothenburg.
Müller, J. R., Isaksson, O., Landahl, J., Raja, V., Panarotto, M., Levandowski, C., & Raudberget, D. (2019). Enhanced function-means modeling supporting design space exploration. AI EDAM, 33(4), 502-516.
Müller, J. R., Siiskonen, M. D. I., & Malmqvist, J. (2020, May). Lessons learned from the application of enhanced function-means modelling. In Proceedings of the Design Society: DESIGN Conference (Vol. 1, pp. 1325-1334). Cambridge University Press.
Ölvander, J., Lundén, B., & Gavel, H. (2009). A computerized optimization framework for the morphological matrix applied to aircraft conceptual design. Computer-Aided Design, 41(3), 187-196.
Raja, V., Kokkolaras, M. and Isaksson, O. (2019), "A simulation-assisted complexity metric for design optimization of integrated architecture aero-engine structures", Structural and Multidisciplinary Optimization, Vol. 60 No. 1, pp. 287–300.
Raudberget, D., Levandowski, C., Isaksson, O., Kipouros, T., Johannesson, H., & Clarkson, J. (2015). Modelling and assessing platform architectures in pre-embodiment phases through set-based evaluation and change propagation. Journal of Aerospace Operations, 3(3-4), 203-221.
Schachinger, P., & Johannesson, H. L. (2000). Computer modelling of design specifications. Journal of engineering design, 11(4), 317-329.
Shergadwala, M., Bilionis, I., Kannan, K. N., & Panchal, J. H. (2018). Quantifying the impact of domain knowledge and problem framing on sequential decisions in engineering design. Journal of Mechanical Design, 140(10).
Stéphane Cueille from SAFRAN on "What does the future hold in store for the Open Rotor?", March 28 2019, available at https://www.safran-group.com/media/what-does-future-hold-store-open-rotor-20190328
Suh, E. S., Furst, M. R., Mihalyov, K. J., & Weck, O. D. (2010). Technology infusion for complex systems: A framework and case study. Systems Engineering, 13(2), 186-203.
Suh, N. P. (1990). The principles of design, 1990. 153, Oxford University Press.
Ulrich, K. (1995). The role of product architecture in the manufacturing firm. Research policy, 24(3), 419-440.
Ulrich, K., & Eppinger, S. (2011). EBOOK: Product Design and Development. McGraw Hill.
Zaporozhets, O., Isaienko, V., & Synylo, K. (2020). Trends on current and forecasted aircraft hybrid electric architectures and their impact on environment. Energy, 211, 118814.

Contact: Massimo Panarotto, Chalmers University of Technology, Department of Industrial and Materials Science, Campus Johanneberg: Chalmersplatsen 4, SE-412 96, Tel: +46 031-7726785, E-mail: massimo.panarotto@chalmers.se