

Analysing increase of functionality and complexity in integrated product architectures

Sabela Gonzalez Castro^{1,2}, Massimo Panarotto³, Olivia Borgue⁴, Ola Isaksson⁵

¹*Chalmers University of Technology*

²*Technical University of Madrid*

sabela@student.chalmers.se

^{3,4,5}*Chalmers University of Technology*

massimo.panarotto@chalmers.se

borgue@chalmers.se

ola.isaksson@chalmers.se

Abstract

Nowadays, product designers find difficulties and uncertainties when assessing how complex a product will be when introducing new functionalities and technologies or when integrating multiple functionalities into a single component. There is a limited number of studies correlating functionality increase to product complexity.

In this paper, the impact of functionality on complexity is quantitatively analyzed in two different case studies: one regarding a set of valves to study how complexity is affected when adding new functionalities, and another one related to three alternative designs of computer mice that features different degrees of functional integration.

In the first case study, it has been found that adding new functionalities leads to a super-linear increase of complexity. In the second case study, it has been found that integrating functionality through the introduction of new electronic elements allows to reduce the total number of components, suggesting a decrease in complexity. However, the case study shows an increase in the product's overall complexity through an increased number of interactions between the components themselves.

This means that possessing a quantitative methodology to evaluate the impact of functionality increase in product complexity can be an important means for designers to avoid misconceptions and risks during the early phases of product development.

Keywords: complexity, functional modelling, integrated architecture product, integrated product development

1. Introduction

Often, product development projects start with the objective of improving the value of an existing product by adding new functionalities (Lindstedt & Burenius, 2006) through new design solutions and components. For example, the functionality of a brake system (e.g. reduce

vehicle motion) can be increased by adding a new function (e.g. recover energy) which is realized by an electric brake. While the introduction of such new functionality is potentially beneficial (e.g. higher energy efficiency in a vehicle); it establishes new links with other parts of the system (e.g. the combustion engine). These new links – if not properly identified and designed – may negatively impact the inherent design complexity of the product, which is often directly linked to product development cost and effort (Sinha, & de Weck, 2013; Raja et al., 2019). One strategy to reduce complexity is to remove unnecessary components (Miles, 1961) through “integrated design” (Pimmler & Eppinger, 1994). In an integrated design, multiple functions are integrated into a single physical component (Ulrich & Seering, 1988). This strategy is often opposed to a “modular design”, with a one-to-one correspondence of functional elements to physical components (Schilling, 2000). Although an integrated design can reduce costs by removing components, it may become constrained in terms of flexibility and functionality for the customer. There is a trade-off between the new functionality that can be introduced in a product and the manufacturer’s need for reducing complexity and ensuring a cost-efficient development. The evaluation of such trade-off in early design is particularly challenging today, since there is not a direct way to link the functionality added in a product with its design complexity.

This research provides a quantitative explanation for how the limited ability to analyse the impact of increasing the functionality in a product – from a complexity standpoint – can become a source of misconceptions and risks. This research focuses in particular on the cases in which the increase of functionality introduces richer dependencies in the product architecture (e.g. integrating electronic, mechanical and information). These design cases are particularly important for manufacturers today, as new advancements in the technological landscape – such as electrification – increasingly enable to enhance the functionality of a product by integrating electronic systems, mechanical parts and information processing into the product architecture (“Evolution of Vehicle Architecture”, 2018).

This paper presents a literature review, where functionality modelling and complexity modelling are introduced. Then, the methodology used and the case studies are presented and detailed. The results obtained as well as reflections and future work are exposed in the discussion. The final conclusion ends this paper with an overall review.

2. Literature review

This section provides an overview of the research regarding functionality and complexity, as well as existing modelling approaches which have been used in the present research in order to quantitatively analyze the impact of functionality increase on design complexity.

2.1. Functionality modelling

In engineering design literature, the *functionality* of a system is the total set of all its *functions*; that is, a singular term to refer collectively to everything it does (Ulrich & Eppinger, 2011). One can say that a machine has certain functions (plural) or a certain functionality (singular). Thus, a function can be defined as “the behavior which is required for the device to satisfy a given requirement” (Pahl, & Beitz, 1984). It has been suggested that the simplest and easiest way to measure the functionality of a product is to count the total number of functions (Griffin, 1993; Kannapan, 1995). However, this way of measuring functionality is considered to be over representative, since a product contains both primary functions (desired by the customer) and “sub-functions” (also called “supporting” and “unwanted” functions; Lindstedt & Burenius, 2006). For these reasons new metrics have been suggested, focusing on the functional decomposition of the product. These new metrics suggest decomposing a product using a functional tree and assume that the greater number of sub-functions, the greater the complexity

(Bashir & Thomson, 1999). This way of looking at functionality assumes that the “true” functionality of the product lies in the primary functions that it provides. These can be seen as the functions that lie in the upper part of the functional tree (also called top-level functions; Bashir & Thomson, 1999). The benefit of looking at top-level functions and sub-functions has made functional trees a popular representation used in industry (Eisenbart et al., 2017) and research has made recent advancements in the functional tree representation. Among others, this research has used an EF-M tree representation (Müller et al., 2019). The EF-M is a formal and 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. The objects used are:

- Functional requirements (FR): states required functions that the system must 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. A DS can be seen the type of solution necessary to solve a FR, thus allowing alternative DS to be represented to fulfil the same FR. As an example, the FR of a steering system can be “control direction of vehicle”. One DS for this FR could be a mechanical steering (e.g. rack and pinion). Another DS for the same FR could be an electromechanical steering (with the use of a hydraulic pump with servo valve). Each of these DSs can be further decomposed into additional FRs and DSs, forming a tree.

The EF-M modelling approach organizes the relations between FRs and DSs 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 as 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, a FR can be realized by only one DS at a time. This generic model allows to define alternative architectures from the very early phases, where quite limited information is available about the forthcoming system.

2.2. Product complexity modelling

In the engineering design literature several product complexity metrics are available, which are based on either physical or functional characteristics (Bashir & Thomson, 1999; Schuh et al., 2004).

Literature is stressing the arduousness of evaluating complexity of products. This is due to the difficulty of separating the single elements of the product, in order to evaluate their complexity. For example, integrated products are defined as opposite to modular products, i.e. it is not possible to split them in components. By integrated architecture products this paper refers to products where the terms, part, component or system take the same meaning and no physical separation is possible (Raja et al., 2019). This creates a difficulty when evaluating the complexity of a product. Moreover, complexity does not only depend on the complexity of the product components, but also on the interactions and interdependencies between components. Pimmler and Eppinger (1995) define four generic interaction types: spatial, energy, information and material. This is shown in Table 1. Besides, the layout of these interactions in the product architecture affects complexity (Sinha & de Weck, 2013). For these reasons, Sinha & de Weck (2013) have defined a metric of structural complexity as:

$$C = C1 + C2 * C3$$

Where C1 represents the sum of complexities of individual components, C2 is a measure of the interactions between components and C3 represents topological complexity (related to the global effect of the arrangement of regions). Table 1 shows and explains the different interactions between elements.

Table 1. Classification of interactions.

N	Flow Type	Description
0	None	No connection
1	Mechanical	Physical coupling between components. This is by nature symmetric.
2	Fluid Flow	Flows of any fluid between two components.
3	Information	Information transfer between components. Generally assumed to be electronic measurement for sensors, etc.
4	Energy	Energy transfer of any energy type.

In this metric, the individual complexity C1 is calculated in function of the TRL i.e., Technology Readiness Level (Mankins, 1995), a type of measurement system used to assess the maturity level of a particular technology (“Technology Readiness Level”, 2012). TRL assesses technology maturity according to a scale from 1 to 9, in which 9 means a level of maturity related to a system proven in environmental operation (European Commission, 2017). According to this model, a component with lower TRL is more complex than a product with higher TRL. C2 measures interaction complexity, and a higher number of interactions implies higher interaction complexity. Topological complexity C3 takes into account the layout of the interactions and increases from centralized towards more distributed architectures (Sinha & de Weck, 2013).

3. Analysing functionality and complexity in integrated product architectures: approach and case studies

This section will introduce the approach used in this research to analyze the functionality and complexity in integrated product architectures. Also, this section will describe the approach in two case studies: one related to diverse valve configurations (featuring different levels of functionality) and one related to alternative integration options in a computer mouse.

3.1. Selected approach to analyse quantitatively functionality and complexity

The approach used to analyze functionality and complexity aims at comparing products based on their capabilities to fulfil functional requirements and their design complexity. Based on the literature review, this research has applied the following approach:

1. Decompose functions and solutions of the design alternatives using an EF-M tree (Müller et al., 2019).
2. Model interactions (*iw*) between components in the EF-M tree, following the classification proposed by Pimpler and Eppinger (1995).
3. Extract DSMs for all the alternatives defined.
4. Apply the complexity metric proposed by Sinha & de Weck (2013) and calculate the final complexity as $C = C1 + C2 * C3$.
5. Count the functionality level of the product by accounting 1) total number of top-level functions *TF* (Bashir & Thomson, 1999) and the total number of functions *N_f*.
6. Plot functionality against complexity.

The following sections will describe how this approach has been applied to two different case studies.

3.2. Case study #1 – adding functions: analyzing the functionality and complexity of valves

The first case study concerns a set of five valves. The authors proceeded first to analyze the “simplest” valve (in terms of smaller number of functional requirements) on a E-FM tree and to modify this model for the other valve configurations by adding the corresponding functions and design solutions. This allows to represent the functions and design solutions involved in each alternative design, as well as to extract the corresponding DSMs.

Figure 1 shows how this modelling activity has been approached. The simplest way to achieve the main function of a valve (“*Regulate the flow of a fluid*”) is by a fixed orifice. To realize this function, the fixed orifice requires three top level functions (“*Inlet fluid*”, “*Outlet fluid*”, and “*Reduce the pressure differential from inlet to outlet*”). Figure 1 on the left shows the E-FM tree of a fixed orifice, together with the corresponding DSM (red cells represents spatial interactions and blue represent fluidic interactions).

However, fixed orifices do not allow the variation of the reduction of pressure differential. To improve the functionality of a fixed orifice valve, a new top-level function is added (“*Calibrate the pressure differential from inlet to outlet*”). This is shown on the right side of Figure 1. To realize this function, the simplest way is to include a mechanical needle. This addition creates new interactions with other elements of the system. These are represented in the E-FM, as well as in the corresponding DSM at the bottom right of Figure 1.

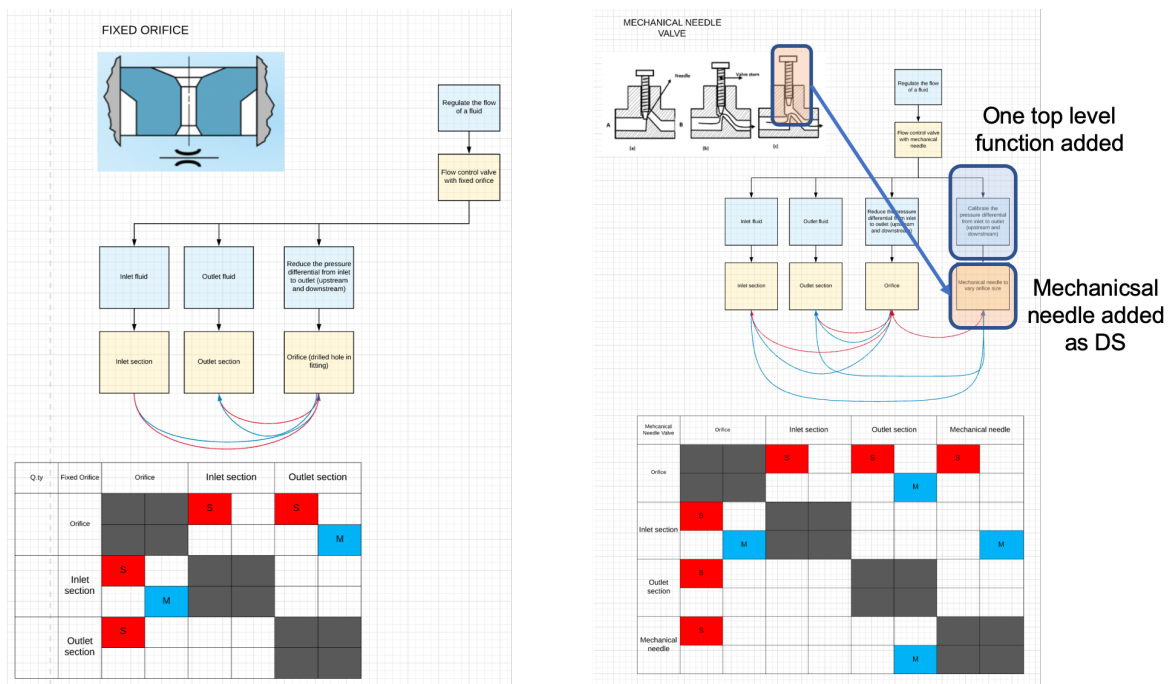


Figure 1. Fixed orifice (left) and mechanical needle (right).

Nonetheless, needle valves do not allow to “*compensate the pressure differential in case of pressure variation in input and output pressure conditions*” (Spiller, 2012). A way to increase the functionality of a needle valve is by adding this top-level function (Figure 2). One design solution to fulfil this top-level function is to design a pressure compensation mechanism (Figure 2). However, this mechanism requires four additional functions, which are solved by four design solutions (e.g. spool) that interact with other parts of the system. This impacts

substantially the DSM shown in Figure 2. Particularly, 2 additional Spatial (S) interactions and 5 extra Material (M) interactions are added.

The analysis is completed by the decomposition of two additional alternatives:

- Pressure and temperature compensated valve: this design extends the functionality of a pressure compensated valve by adding a temperature compensator that corrects the orifice opening in function of the viscosity of the flow, which is affected by temperature. In terms of DSM, this adds a new element (a “Temperature sensitive element”) which creates a new Energy (E) interaction in the DSM.
- Solenoid pressure and temperature compensated valve: performs variable control of the flow despite temperature and pressure changes. It is controlled electromechanically, what offers more sensitivity and quicker response. In terms of DSM, this changes 1 element (“spool” instead of “needle”) and adds 3 new elements (“On off solenoid”, “Input power generator” and “Control spring”), what supposes 4 new Spatial (S) interactions and 1 new Energy (E) interaction.

After decomposing these five valve alternatives, they were analyzed in terms of complexity. Table 2 shows the results obtained from the functional modelling and after applying the structural complexity metric to the set of valves. From Table 2, it can be seen that when the design alternatives include more functionalities it also implies a greater number of design solutions, top level functions and interactions, what traduces into a higher structural complexity.

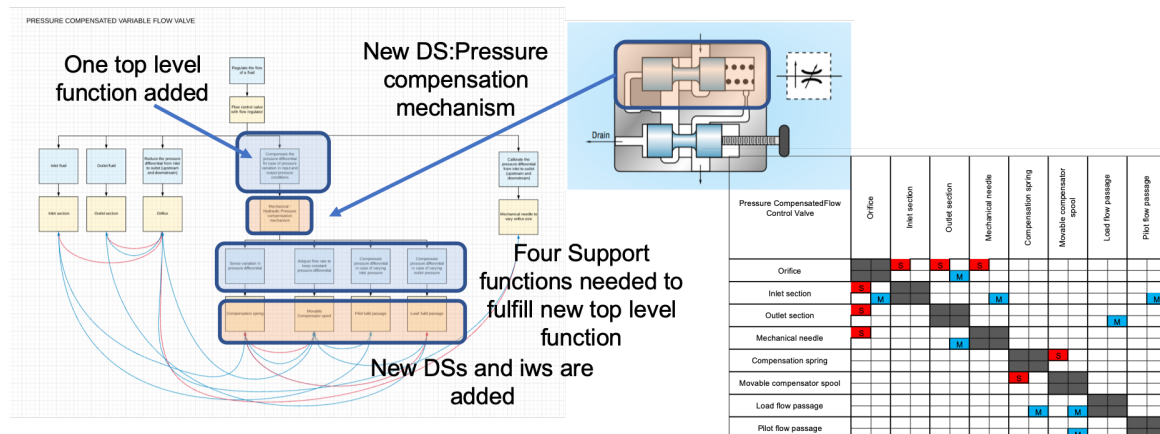


Figure 2. Decomposition of pressure compensated valve.

The topological complexity C3 is the only one that does not always increase, as it is showed. This is explained by its own definition, because it is attributed to the specific layout of the system under consideration (Raja et al., 2019). In Table 2, it can be seen that for the valves that have higher C3, interactions are more distributed and adopt a less compact structure. The analysis focused on spatial, material and energy interactions. Information links were not included because they were zero for all of the designs. The results obtained have been represented in Figure 3 in order to reflect on complexity.

Figure 3 shows the relation between complexity and number of top-level functions for each type of valve.

Table 2. Results obtained for the set of valves.

	Fixed Orifice	Mechanical Needle	Pressure Compensated Flow Control Valve	Pressure and Temperature Compensated Flow Control Valve	Solenoid Valve Pressure and Temperature Compensated
DSM dimension	3x3	4x4	8x8	9x9	12x12
Top-level functions	3	4	5	7	8
Total functions	3	4	10	12	17
Spatial links	4	6	8	10	14
Material links	2	4	9	9	9
Energy links	0	0	0	1	2
Total links	6	10	17	20	25
Connection density (%)	66.7	62.5	26.6	24.7	17.4
Components complexity C1	0	1.9	14.4	17.5	26.9
Interface complexity C2	3	5	8	9	12.5
Topological complexity C3	1.5	1.5	1.2	1.4	1.2
Structural complexity C	4.5	9.5	24.3	30.2	41.8

To illustrate the implications of these results, this research considers a scenario in which the company has already developed the first two designs (fixed orifice and mechanical needle, green area of Figure 3) and where designers are assessing the impact of the other three alternatives (red part of figure 3) in order to decide about which design to develop.

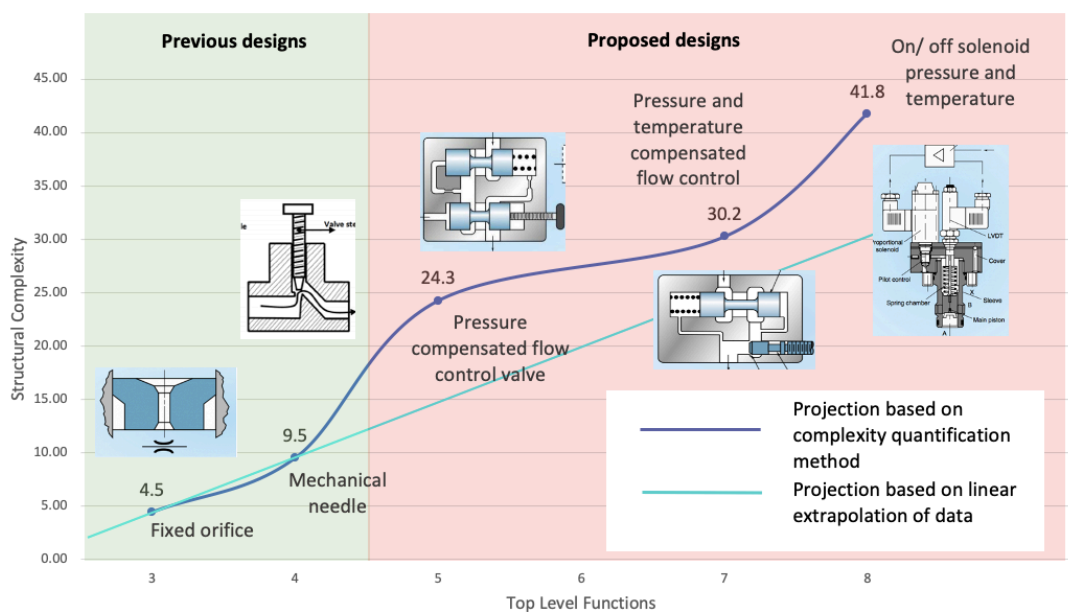


Figure 3: representation of number of top-level functions according to complexity.

Without the proposed approach for functionality and complexity analysis, engineers could just apply a linear approximation based on the already developed designs. Figure 3 shows how such linear approximation could lead to an underrepresentation of complexity. For example, the

structural complexity correspondent to a pressure compensated flow control valve (with 5 top level functions) would be of around 15, instead of the value of 24.3 obtained with the present methodology. This means an error of 39.8%. At the same time, applying a super-linear relation between functionality and complexity could lead to misconceptions as well. This is because complexity depends on the design solutions adopted to fulfil the functions. For example, a pressure and temperature compensated flow control valve implies only a slight increase in complexity compared to the pressure compensated valve (from 24.3 to 30.2), whereas two top-level functions are added (from 5 to 7). Conversely, improving the temperature compensated valve with the addition of only one more function (featuring a fine calibration of temperature with a solenoid valve) implies a more substantial increase in complexity (from 30.2 to 41.8). This is because a more complex set of energy interactions (constituted by electrical components and interfaces) are introduced with the solenoid valve.

These results show how complexity is depending on the “criticality” of the function from a complexity standpoint, and it depends on the type of design solution adopted.

3.3. Case study # 2 – integrating functions: analyzing functionality and complexity of different computer mouse configurations

In this case study, three computer mouse models are considered. The main difference compared to the valves is that, instead of introducing new top-level functions, the mice models present the same top-level functions. What differs is the level of integration of these functions into design solutions. The first “old model” is a common mouse that has three switches at the upper case: left, right and the wheel in the middle. This “old model” is shown in the left part of Figure 4. Internally it has a mechanism that allows to detect and create signals for movement. These switches will make contact with three internal sensors when the user decides to select. The next alternative design is a “tactile” one. The upper case only presents one switch because inside the case a set of capacitive touch sensors were included. Thus, the tactile mouse performs the same number of top-level functions as the old mouse, but with a smaller number of components instead. Hence, the tactile mouse is a more integrated design since, thanks to electronics, each component or subsystem performs, i.e. integrates, more functions.

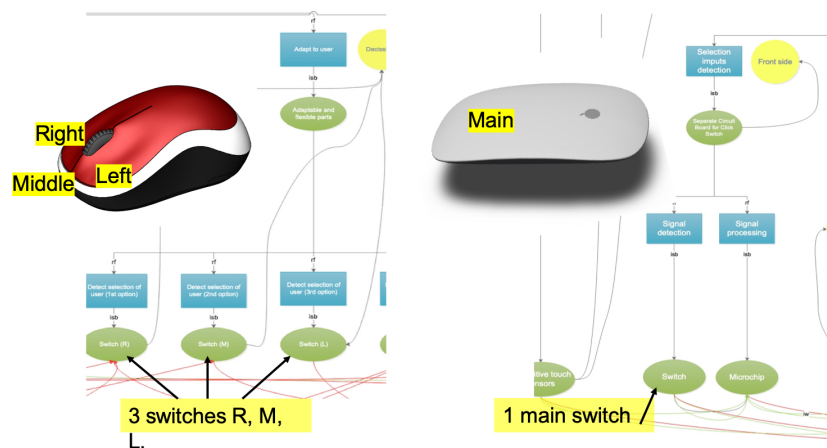


Figure 4: Comparison between two first mice with three switches and tactile mouse with one switch.

The substitution of several mechanical elements for electronics is an interesting issue and the aim is to establish if this action increases the complexity of the system or not and how. This can be seen in the EF-M trees created and shown in Figure 5. Compared to the valve cases, spatial, energy and information interactions are considered. Material links do not apply in this case.

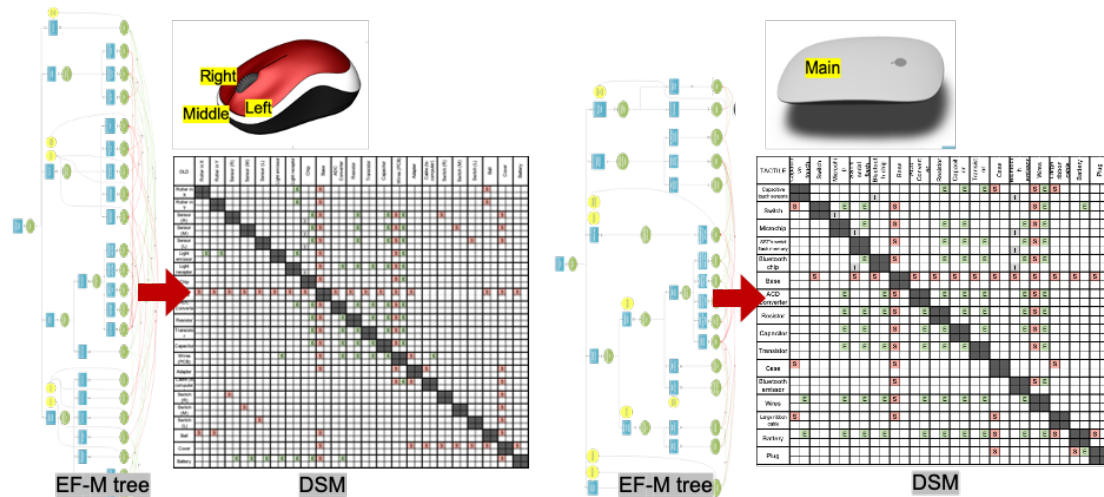


Figure 5: EF-M tree and DSM obtained for the old and tactile mouse models (red and green cells correspond to spatial and energy interactions, respectively).

As exposed before, integrated product architectures are difficult to analyze since several functions are integrated in one single element. This is why functional modelling is an important tool to study these products. Breaking down each integrated product in functions and solutions (through the EF-M trees and DSMs) allows to consider different functionalities, relate them with design solutions, study the interactions between these and have the capability to analyse how to enhance them.

Table 3 shows the results obtained from the EF-M trees decomposition and application of the complexity metric. An intermediate mouse model was also studied. The difference with the “old mouse” is that, instead of a cable, it is wireless and the battery rechargeable.

Table 3. Results obtained for the set of mice.

	Old: Wire, non-rechargeable	Enhanced Wireless + rechargeable	Tactile + Wireless + rechargeable
DSM dimension	22x22	23x23	16x16
Number of components	22	23	16
Top-level functions	6	6	6
Total functions	29	31	24
Spatial links	64	62	32
Energy links	59	61	70
Information links	4	5	6
Total links	127	128	108
Connection density (%)	26.2	24.2	42.2
Components complexity C1	0	6.3	18.8
Interface complexity C2	70	70	65
Topological complexity C3	1.9	1.9	2.1
Structural complexity C	128.5	136.7	159.3

Table 3 shows how the number of design solutions and components of the tactile design is considerably smaller (16 instead 22). Not only the switches but also the internal mechanism (that contains several mechanical elements) for detecting movement was also changed for a smaller number of electronic elements. After removing the mechanical elements present in the first mouse, the number of components and total functions decreased, comparing both old and tactile mice. Spatial links or interactions were reduced by half (32 from 64), while energy and information links were increased, but considerably less than the reduction. The connection density increased since the dimension of the DSM considerably decreased. Looking at the table it is seen that between the first and intermediate mice, the number of components increased. The reason is that the cable and adaptor were removed (2), but an emitter, transmitter and plug were included (3). Regarding the complexity calculations and starting with the individual complexity of each component, it was considered that for the first “old” mouse model the TRL of the components is 9, which means that they are proven in operational environment, and makes complexity equal to zero. This complexity increases when introducing new elements as the emitter (instead of the wire) and the capacitive sensors (instead of the elaborated mechanisms) because of the lower TRL. The new elements permit to reduce the number of “unnecessary” functions, which is shown in Figure 6 (left).

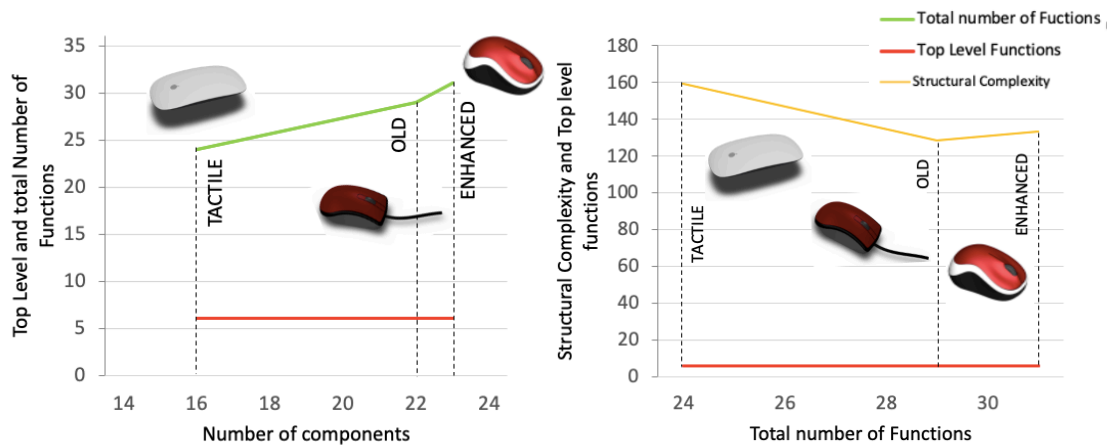


Figure 6. Relation between functionalities and components (left) and relation between functionalities and complexity (right).

Interface complexity reduces with enhancements because several elements are deleted and interactions, as exposed, decrease. Topological complexity increases because the total number of links, although is reduced, does not considerably change, and with a smaller number of components the density and layout are more elaborated. It is found that complexity increases with new integrated design alternatives, but the reason is mainly the individual complexities of each component. Nevertheless, if the number of components and total number of functions are compared, it is seen that the newest tactile design requires a smaller number of components, as it is shown in Figure 6 (left). The number of components is often related to the material cost (Raja et al., 2019), higher for the old mouse design, whereas complexity often relates to the product development cost (Sinha, & de Weck, 2013), higher for the tactile design. Hence, the fact that the newest electronic design reduces the number of components could mean a smaller product cost but, as a trade-off, increases the total complexity, what could also lead to a higher product development cost.

4. Discussion

The case study of the valves shows that considering a linear relation between the complexity and the number of functionalities added in a product can lead to a misconception. In the case

study, this misconception reaches an error of 40%. Regarding the case study of the mice, it is shown that removing mechanical elements and substituting the mechanism with electronic components allows to reduce the number of components and function requirements for the same number of top-level functions, suggesting a reduction of product complexity. However, improving the mouse design with such an integrated product strategy (through “function sharing”, Ulrich & Seering, 1988) means actually to increase the overall design complexity. While the product features less components, the new interactions introduced are affected significantly.

This research has provided a quantitative explanation for how the limited ability to analyse the impact of increasing the functionality in a product architecture can become a source of misconceptions and risks. This study suggests that an approach correlating the impact of functionality increase can be helpful and effective to design next-generation product architectures (characterized by a high degree of technology blending). In this regard, design solutions are also carrier of non-functional aspects, which have to be considered and traded in early design. This approach supports this need of considering non-functional requirements, since complexity can be considered as one functional requirement (Raja et al., 2019).

The current approach applied in this paper has a number of limitations which require further attention in the future. In particular, the results of the complexity metric suggested by Sinha & de Weck (2013) are dependent on the EF-M tree and DSM decomposition, as well as on the assignments of the complexity of each interaction (e.g., the TRL of each component). These judgements involve a considerable level of subjectivity. Future work will focus on removing these sources of uncertainty by using data coming from simulations to populate the DSMs and the complexity metric. For example, Raja et al. (2019) carried out an analysis to study spatial design solutions and interactions using *Parametric Connecting Graphs* (PAG) and Load Path Analysis. Future work will focus on extending this approach considering the complexity of electric, fluidic and software interactions using simulations belonging to each respective domain. Moreover, the complexity of a product has a strong impact in quality, process lead time and every stage of a product development process. Hence, future works will aim for a solution that quantifies how new technologies and innovations affect these and how to improve them.

5. Conclusions

The purpose of this paper was to study how increasing functionalities leads to complexity in integrated products. The selected approach was based on functional modelling (EF-M trees), DSMs and a metric for complexity derived from literature. Two case studies were analyzed, which represent two common design situations: adding functions to an existing product and integrating functions into a single component. The results of these studies provide a quantitative explanation for how the limited ability to analyse the impact of increasing the functionality in a product architecture can become a source of misconceptions and risks.

The results point to the benefit of applying a methodology that allows to correlate functionality with complexity can be helpful and effective when designing next-generation product architectures (characterized by different degrees of integration and a high degree of technology blending).

Future work will focus on improving the proposed methodology by using data coming from simulations to populate the DSMs and the complexity metric. Also, future work will focus on validating the proposed method with real industrial case studies. Current work is ongoing in collaboration with a manufacturer of fluid management systems for satellites. The planned approach for validation is to correlate the complexity metrics to the product development costs of products already developed by the manufacturer, in order to validate the correlation between complexity and development costs. Sinha & de Weck (2013) have also followed this approach.

Citations and References

- Bashir, H. A., & Thomson, V. (1999). Estimating design complexity. *Journal of Engineering Design*, 10(3), 247-257.
- Eisenbart, B., Gericke, K., & Blessing, L. T. (2017). Taking a look at the utilisation of function models in interdisciplinary design: insights from ten engineering companies. *Research in Engineering design*, 28(3), 299-331.
- Evolution of vehicle architecture. (2018, June 21). Retrieved from <https://www.aptiv.com/newsroom/article/evolution-of-vehicle-architecture>
- Griffin, A. (1993). Metrics for measuring product development cycle time. *Journal of Product Innovation Management: An International Publication of the Product Development & Management Association*, 10(2), 112-125.
- Kannapan, S. M. (1995). Function metrics for engineered devices. *Applied Artificial Intelligence an International Journal*, 9(1), 45-64.
- Lindstedt P, Burenus J., 2006. The Value Model: How to Master Product development and Create Unrivalled Customer Value. Nimba, Sweden.
- Mankins, J. C. (1995). Technology readiness levels. White Paper, April, 6, 1995
- Miles, L.D. (1961), *Techniques of Value Analysis and Engineering*, McGraw Hill, New York.
- 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.
- Pahl, G. & Beitz, W. (1984) *Engineering Design: A Systematic Approach* (London, Design Council).
- Pimmler, T. U., & Eppinger, S. D. (1994). Integration analysis of product decompositions.
- Raja, V., Kokkolaras, M., & Isaksson, O. (2019). A simulation-assisted complexity metric for design optimization of integrated architecture aero-engine structures. *Structural and Multidisciplinary Optimization*, 60(1), 287-300.
- Schilling, M. A. (2000). Toward a general modular systems theory and its application to interfirm product modularity. *Academy of management review*, 25(2), 312-334.
- Sinha, K., Omer, H., & de Weck, O. L. (2013). Structural complexity: Quantification, validation and its systemic implications for engineered complex systems. In DS 75-4: Proceedings of the 19th International Conference on Engineering Design (ICED13), Design for Harmonies, Vol. 4: Product, Service and Systems Design, Seoul, Korea, 19-22.08. 2013 (pp. 189-198).
- Spiller, D. (2012). *Engineering Essentials: Flow-Control Valves*. Retrieved from <https://www.hydraulicspneumatics.com/technologies/hydraulic-valves/article/21885085/engineering-essentials-flowcontrol-valves>
- Suh, N. P. (1990). *The principles of design* (No. 6). Oxford University Press on Demand.
- Schuh, G., & Eversheim, W. (2004). Release-Engineering—An Approach to Control Rising System-Complexity. *CIRP Annals*, 53(1), 167-170.
- Technology Readiness Level. (2012, October 28). Retrieved from https://www.nasa.gov/directorates/heo/scan/engineering/technology/txt_accordion1.html
- Ulrich, K. T., & Seering, W. P. (1988, August). Function sharing in mechanical design. In *AAAI* (Vol. 88, No. 1, p. 342).
- Ulrich, K., Eppinger, S., 2011, *Product design and development* (5th edition). New York: McGraw-Hill.
- Andreasen MM (1980) *Syntesemetoder pa Systemgrundlag - Bidrag til en kon- struktionsteori*. (PhD Thesis), Lund University.