



EXTRAPOLATION OF DESIGN GUIDELINES DURING THE CONCEPTUAL DESIGN PHASE: A METHOD TO SUPPORT PRODUCT ARCHITECTURE DESIGN

G. Formentini^{1,✉}, C. Favi¹, F. Bouissiere², C. Cuiller², P.-E. Dereux², R. Guillaume² and C. Malchair²

¹ University of Parma, Italy, ² Airbus S.A.S., France

✉ giovanni.formentini@unipr.it

Abstract

The work aims at the definition of a design methodology able to drive designers in the definition of product architectures, starting from rough information available at the conceptual design. The methodology identifies design guidelines useful for the development of product architectures optimized for a given target (i.e. assembly, cost). The methodology is based on domains which combine attributes related to a specific aspect of the target. The exploratory application of the methodology was performed to address the equipment installation of a civil aircraft for assembly/installation target.

Keywords: *conceptual design, design guidelines, product architecture, optimisation, design methodology*

1. Introduction

In today's competitive business environment, companies are trying to constantly innovate processes and products with the aim to reduce costs and to increase product quality and productivity; indeed the optimization is mainly performed during the product development process, with the aim to achieve specific targets, such as the reduction of assembly time, the improvement of product modularity, the optimization of manufacturing process, etc.

The formalization of product development process has been proposed by (Pahl et al., 2007) where four (4) phases have been identified: (i) *Planning and Task Clarification*, (ii) *Conceptual Design*, (iii) *Embodiment Design*, and (iv) *Detail Design*.

Optimizations are generally implemented during the latest phases of the design process. Several methods, such as DFMA (Design for Manufacturing and Assembly), DtC (Design to Cost), etc., were developed in the early 80's to improve different targets starting from virtual product models or prototypes. All these methods are collected under the name of Design For X (DFX), where the "X" considers the target to be optimized (Pahl et al., 2007; Otto and Wood, 2001).

One of the most popular DFX methods is the DFA (Design for Assembly) developed by Boothroyd et al. (2010) where a set of empirical charts are used to estimate the assembly time of different products. The method relies on the definition of two assembly operation categories: handling and insertion (Namouz et al., 2012). The method allows a reduction in part numbers and simplification of assembly operation, but it requires having a product prototype (virtual or physical) which means the method can

be applied only at the *Embodiment Design* phase or later at the *Detail Design* phase. However, the effect of design actions in the early phase of design (i.e. *Conceptual Design*) allows engineers and designers to drastically change the products and level of technology within the product. The *Conceptual Design* phase is the one in which the principle solution of the product under analysis is defined, and where the degrees of freedom in design space solutions is higher (Pahl et al., 2007). The first attempt to anticipate design actions oriented to product assembly at the conceptual phase was proposed by Stone et al. (2004). The authors proposed to start the DFA with the functional definition of the product, then, applying the developed heuristic method, to derive product modules (i.e. groups of sub-functions) that can be optimised from the assemblability point of view. The method by Stone et al. (2004) is very promising in the development of products characterized by the principle of modularity. A modular product is defined as a product which present sub-systems that can be clustered as a unit (module) and that serves identifiable function (Dahmus et al., 2001). The method presented by Stone et al. (2004) presents some drawbacks reported in the literature: (i) the assembly sequence is not considered within the method (Favi and Germani, 2012), (ii) the method applicability is limited for small and fully modular products (Bouissiere et al., 2019) and, (iii) design suggestions are not provided to practitioners (Demoly et al., 2011). Concerning the first two issues, few attempts were done to overcome these limitations. For example, in the work of Favi and Germani (2012), the authors proposed to identify a proper assembly sequence starting from the interface analysis. Bouissiere et al. (2019) provided a method to assess, analyse and compare different product architectures, tackling a very challenging product: an aircraft. A product architecture is defined as the system representation showing the relation between product function structure and its physical configuration (Pahl et al., 2007). Ulrich and Eppinger (2011) defined product architecture as the scheme where every function of the product is clustered into physical components. Usually, the product architecture is used to develop modular products by clustering into groups different functions, that will make a customizable and replaceable module. In order to achieve a modular product different tools can be adopted, such as Design Structure Matrix (DSM), (ii) Modular Function Deployment method (MFD), Quality Function Deployment method (QFD), etc. (Fiorineschi et al., 2015). The reported methods were developed to be used at the *Conceptual Design* phase aiming at analysing product architectures for the definition of modular products. These methods are considered valid tools in the analysis and comparison of product architectures; however, they lack in the definition of specific design guidelines that can drive designers in the definition of product architectures with optimized features in terms of manufacturing and assembly. This is the main gap observed by the analysis of the literature about this topic.

This paper aims at the characterization of design guidelines that can be used at the *Conceptual Design* phase for the development of modular products concerning specific targets (i.e. fit for assembly performances). The proposed work provides a method to derive design guidelines able to drive product architects and engineers in the definition of optimized product architecture solutions. The method consists of several steps: (i) it starts with the definition of the aspect to be improved (e.g. product fit for assembly performances, etc.), (ii) it analyses the current product classifying target-related parameters (attributes), (iii) it identifies criticalities and positive aspects related to the attributes with the definition of scoring matrices, (iv) it combines attributes with mathematical equations by the adoption of hierarchical structure, (v) it assesses the critical aspects (modules, connections, etc.) related to the given configuration and, (vi) it extrapolates design guidelines based on main issues identified with the assessment. The hierarchical structure is the core of the method; it provides a way to explore the results through different domains and levels to harvest information, useful to create the design guidelines applicable at the conceptual phase of the design process. The paper is structured as follow: after this introduction (1), the *Materials and Methods* section reports the proposed approach as a whole (2). The *Case Study* section provides an evidence of the extrapolation of design guidelines related to the fit for assembly of an aircraft component (i.e. the nose fuselage) (3). *Results* obtained by the application of the proposed approach in the specific case study (4) and *Conclusions* (5) sections are reported as well.

2. Materials and methods

Before introducing the method, it is necessary to understand which information and tools are available at the *Conceptual Design* phase. During the early phases of the design process, the solution space is

extremely wide, and the available information is very rough, indeed it is obtained from the product requirements (Golob et al., 2002). In order to best use the available information, it is necessary to structure data in a proper way. Moreover, tools such as functional representation and digital mock-up (DMU) can be used to support the harvesting of information. The functional representation allows to graphically represent functional modules and connections among modules with different level of granularity and with a standardized coding. The *simplified* digital mock-up (Boussiere et al., 2019) is a tool that allows users to create a simplified version of the desired product by designing it in a virtual environment and using simple features (i.e. cylinders, bricks, etc. that represent by the bounding boxes of modules and connections). The designers can quickly create a virtual prototype of the product containing relevant geometrical information without precisely mirroring the reality. Knowing this, the proposed method consists of six (6) steps (Figure 1).

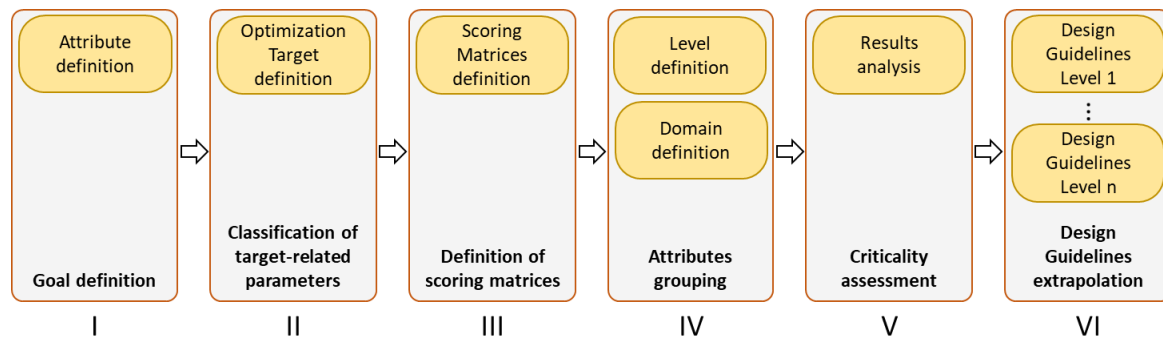


Figure 1. Methodology steps

The core of the methodology is the creation of a hierarchical structure of *attributes*, *domains* and *levels*. This is a crucial aspect since this type of structure will allow to identify the main contribution of each considered item to the final score and to use this information to generate specific guidelines for the improvement of product architectures. The first step of the approach is the **Goal definition (target)**. It allows to focus on the specific target to improve (i.e. reduce overall assembly complexity). Once the aim of the method is defined, then, it is necessary to identify goal related attributes.

Classification of target-related parameters (attributes) is the second step of the proposed approach and it allows to characterize all possible aspects related to the given target. Knowing the optimization target (i.e. reduce overall assembly complexity), it is required to analyse the current available product architecture for the definition of *attributes*. An attribute is a parameter that influences the chosen target. Information about the derived attributes must be available at the *Conceptual Design* phase otherwise it will not be possible to proceed with the analysis. For example, if the module geometry/dimension is identified as an attribute, the latter must be considered using the available information at the *Conceptual Design* phase. In this case, the exact geometry of modules/components is not available, so the bounding box is used as numerical parameter for the module geometry/dimension. The definition of attributes allows to understand link between information available at the *Conceptual Design* phase and the methodology's target. In the case of the target of the analysis is the reduction of overall assembly complexity, the attribute "available working space for operators" is a key attribute to assess the target, and by the observation of the current situation at the assembly floor, it is possible to understand issues and positive feedbacks associated to this attribute.

Definition of scoring matrices is the third step and it allows to identify positive and negative features related to a given attribute. For each attribute, data from functional analysis and DMU are collected, then, based on the attribute's typology (i.e. string, numbers, percentage, etc.), a normalization process is required. Thus, the use of *scoring matrices* allows to have homogeneous data (numbers) to perform mathematical operations. A *scoring matrix* is defined as a table which translates target features in numerical values (numbers in a range between 1 and 5). The matrix is composed of two columns: (i) range of a specific feature, and (ii) score (Favi et al., 2019). Once attributes and related scoring matrices are defined, they need to be clustered combining the attributes that have common features. The creation of attributes groups requires the adoption of a hierarchical structure and the implementation of mathematical equations to combine attributes properly. **Attributes grouping** is the

fourth step of the proposed approach and it allows to define interdependencies among attributes creating a hierarchical structure by applying the right mathematical model to describe the objective of the analysis. For example, if the considered target of the analysis is the reduction of overall assembly complexity, “available working space for operators” and “working position” are related attributes that can be clustered within the same *domain* and with a simple mathematical operator (i.e. sum, root mean square). A *domain* is defined as a group of attributes related to specific aspects. Domains are defined based on the final goal of the method. Domains interact with each other at different *levels*; thus, it is required to define *levels* where domains are clustered together (Figure 2).

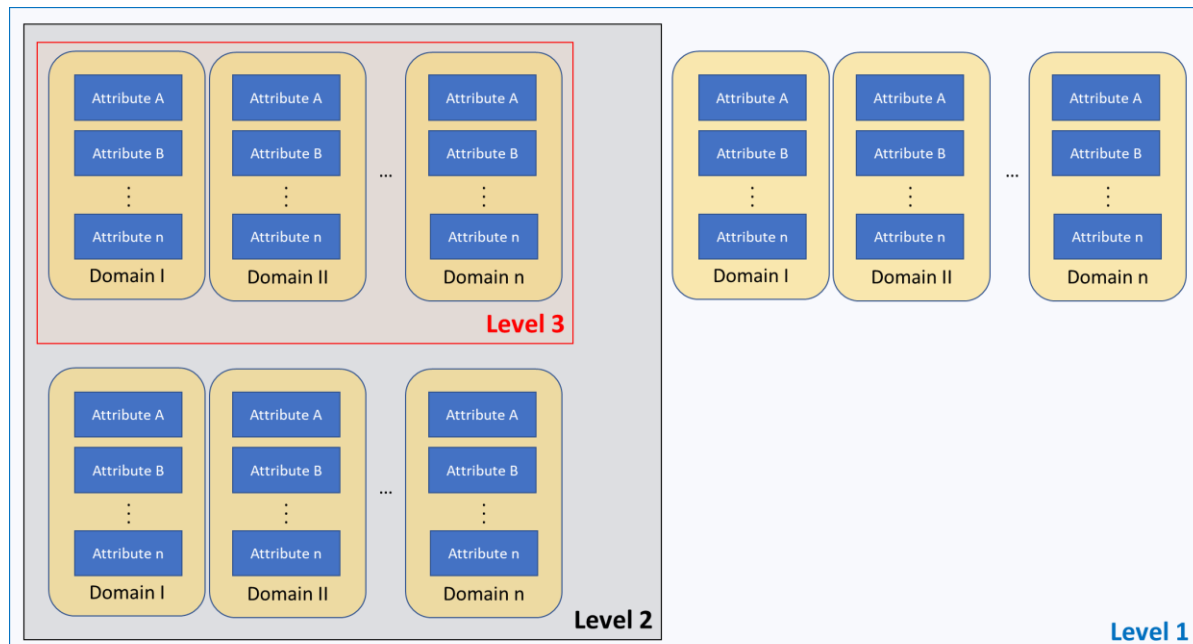


Figure 2. Hierarchical structure

A more detailed description of *domain* definition, *levels* and mathematical equations are provided in a previous work from the same authors (Bouissiere et al., 2019). Information collected inside attributes are processed with specific mathematical algorithms that: (i) normalizes information with the scoring matrix approach (Favi et al., 2019), (ii) collects all normalized scores into a single value, ranging from 0 to 1 (Bouissiere et al., 2019). The numerical assessment has three different purposes: (i) it provides a global understanding of a given architecture related to the target of the analysis, (ii) it addresses the most critical items inside a given architecture and, (iii) it defines a benchmark to compare alternative solutions and new product architectures. **Criticality assessment (modules, connections, etc.)** is the fifth step of the proposed approach and it allows to gather design information about what to do and what to avoid in a given product architecture. The analysis of critical items starts from the highest scores (i.e. the most critical five scores) and continue through the navigation of the hierarchical structure to understand the rationale behind their score: (i) levels, (ii) domains and, (iii) attributes. Information becomes more specific as levels are increased (i.e. information in *Level 1* has a higher generality than information in *Level 2*, etc.). The designer can choose at which level the analysis needs to be stopped and, from the information stored at that level, define general design guidelines (referring to a domain) or specific design guidelines (referring to an attribute). **Design Guidelines extrapolation** is the final step and it allows to guide designers in the definition of new architectures. Design guidelines are used to make a re-design for the optimization of a given target or to develop a brand-new architecture. The definition of design guidelines is obtained with the following syntax: (1) *verb* - it expresses the action to perform; (2) *name* - it can be the name of the attribute related to the domain in the level of interest, the name of the module or the name of interface that is analysed. The choice depends on the level of granularity the design guidelines is associated to. (3) *constraint* - it expresses the core of the guidelines. It can be a threshold (i.e. number, range of numbers, etc.) or a particular requirement (i.e. an area to avoid, etc.) (Figure 3).

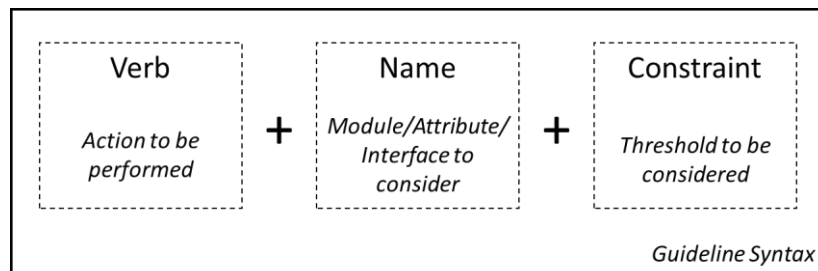


Figure 3. Design guidelines syntax

The derivation of design guidelines can be performed at different levels of granularity. First, the *Level 1* is analysed, and general guidelines are derived from it. It is important to notice that design guidelines referred to this level are easy to assess and, generally, in accordance with the common sense and engineering judgement. This is due to the generality of the information contained in the first level. The extrapolation of design guidelines continues with the analysis of the second level. As the level information's detail increases (i.e. more specific) the derivable design guidelines become more product related. Indeed, by analysing the *Level 2* and *Level 3*, design guidelines related to the product under development are obtained. Finally, the last analysis can be performed on the scoring matrices. By analysing the scoring matrices used to run the method, it is possible to derive useful design guidelines which refer at each single attribute considered in the analysis. Senior designers might already possess this information, but it is very useful to junior designers and/or new architects who are not familiar with the product. The benefit of the proposed method is the possibility to grasp the way in which the combination of attributes leads to specific results and, from these results, derive different design guidelines. The design guidelines derived by the adoption of the proposed method would not have been identifiable only with a collection of the company-knowledge. Indeed, company knowledge is used as ground information while the mathematical models combine the available knowledge to describe the target model, with the aim to define the impact on the product architecture. In addition, all the derived design guidelines can be stored in a company repository and shown to the user to guide the modification of modules and/or connections aiming at improving the product architecture towards the desired goal.

3. Case study

The proposed approach was applied in the context of product assembly/installation, analysing a complex assembly such as the nose-fuselage of an Airbus aircraft (Figure 4). Indeed, the nose-fuselage is one of the most complex part of the aircraft, it presents many elements (electronic equipment, air pipes, etc.) that need to be installed in a very confined area and its architecture is constrained by several aspects. This level of complexity is considered a valid testbed for the application of the proposed approach.

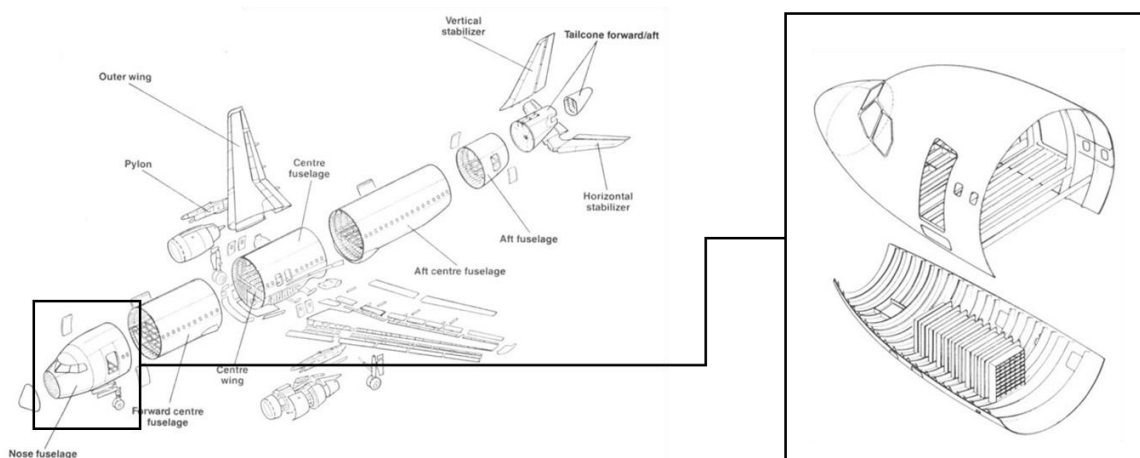
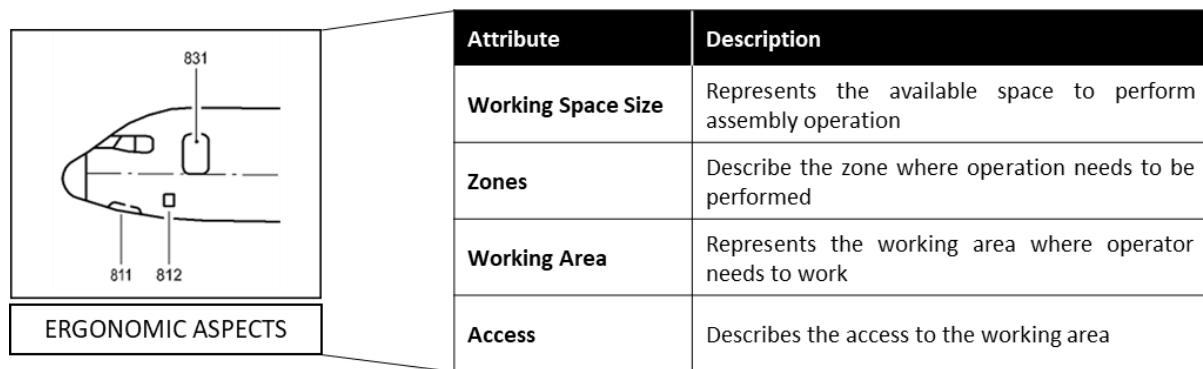


Figure 4. Nose fuselage structure

The aim of the methodology was to identify design guidelines in order to reduce the assembly complexity at the conceptual level. The case study was analysed following the approach described above.

The first step was the **Goal definition (target)**. In this case, the goal of the analysis is to develop optimized product architecture for nose-fuselage assembly reducing the overall assembly complexity, both in terms of critical operations and time-related process performance. In this case, the installation and assembly of modules (equipment, devices, etc.) and interfaces (pipes, ducts, cable, etc.) are considered in the critical path for the improvement of product capacity.

The second step was the **Classification of target-related parameters (attributes)**. The attributes identification was carried out through three (3) meetings between Engineering and Manufacturing departments of AIRBUS Company. During the meetings, the main installation-related challenges were tackled, properly characterizing the related attributes. For the sake of brevity, here below is reported the example of the definition of the ergonomic domain starting from the main attributes that have been derived to consider ergonomic aspects: (i) Working Space Size, (ii) Zones, (iii) Working Area and, (iv) Access. Each attribute is detailed in Figure 5. Ergonomic is only the first collection of attributes in a given domain. For this case study, four different groups of attributes were addressed: (i) ergonomic aspects, (ii) assembly operation, (iii) components topology, and (iv) interface type. This set of data was then used in the fourth step of the methodology to create assembly “domains”.

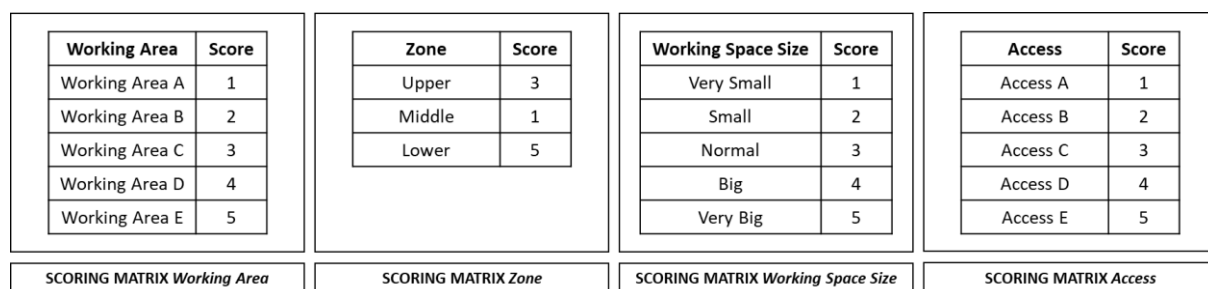


Attribute	Description
Working Space Size	Represents the available space to perform assembly operation
Zones	Describe the zone where operation needs to be performed
Working Area	Represents the working area where operator needs to work
Access	Describes the access to the working area

Figure 5. Attributes identified for the “ergonomic domain”

The third step was the **Definition of scoring matrices**. For each identified attribute, a scoring matrix was defined. The definition of a scoring matrix was carried out with a concurrent design approach between manufacturing and architectural departments. The definition of scoring matrices allows to: (i) provide a better understanding of the identified attributes, releasing the ones not related to the study, (ii) translate all information in a mathematical form, and (iii) formalize the company knowledge.

Again, for the sake of brevity, the scoring matrices identified for the attributes related to ergonomic aspects are shown in Figure 6. It is worth noting that scoring matrices are basic information that characterizes the assembly matter. Hence, the scoring matrices are independent from the model that is used to describe the assembly problem.



Working Area	Score
Working Area A	1
Working Area B	2
Working Area C	3
Working Area D	4
Working Area E	5

Zone	Score
Upper	3
Middle	1
Lower	5

Working Space Size	Score
Very Small	1
Small	2
Normal	3
Big	4
Very Big	5

Access	Score
Access A	1
Access B	2
Access C	3
Access D	4
Access E	5

SCORING MATRIX Working Area SCORING MATRIX Zone SCORING MATRIX Working Space Size SCORING MATRIX Access

Figure 6. Scoring matrices for the attributes of “ergonomic domain”

The fourth step was the **Attributes grouping**. To define proper *domains* and *levels*, a deep understanding of the product is required. Domains were created from the hypothesis that there are four aspects that always interact during the assembly process. Four domains were identified: (i) Ergonomic Domain, (ii) Assembly Domain, (iii) Component Domain and, (iv) Interface Domain. Subsequently, three levels were created in order to link all domains properly. The *First Level* contains only the *Interface Domain*; indeed, this domain presents the highest level of granularity. The *Second Level* contains *Ergonomic* and *Assembly Domains*. These two domains present information that are related to the assembly operation of aircraft modules (i.e. manual operations in a given space or working area, tools used for module installation, etc.). Finally, the *Third Level* incorporates only the *Component Domain* because represents the deepest level of granularity (i.e. specific information related to the piece to install such as 3D shape, rough size, rough weight, etc.). The hierarchical structure used is shown in Figure 7. Inside each domain, information of all attributes is collected with the use of specific mathematical operators (i.e. mean operator, RMS operator, etc.). For example, normalized values inside the Ergonomic Domain have been collected with the Root Mean Square (RMS) operator (Figure 8).

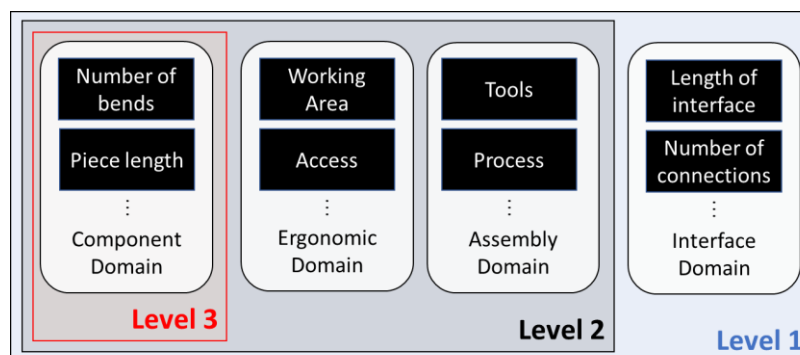


Figure 7. Hierarchical structure of the assembly problem

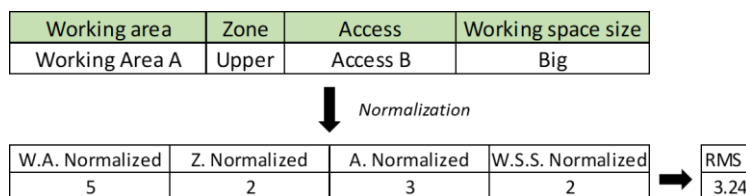


Figure 8. Ergonomic domain values collection

The fifth step was the **Criticality assessment** (*modules, connections, etc.*). This task was performed following the overall framework described above and the results for modules were ranked from highest to lowest score (Figure 9). The purpose of this assessment is to identify the most critical modules in order to focus the design efforts towards the most critical items of a given configuration and iteratively progress towards a global optimum.

The last step was the **Design Guidelines extrapolation**. To derive guidelines properly, the Level 1 was tackled first. By doing so, it was possible to obtain high-level design guidelines related to interface analysis. For example, modules characterized by a high score have an elevated number of interfaces, thus, guidelines such as, *keep number of interfaces below a threshold J* or *split interface length above a threshold M* were recorded. Indeed, *Level 1* accounts aspects related to interfaces that are connecting modules together. Reducing scores of attributes inside the *Level 1* will affect the interface installation and connection to the module itself, avoiding assembly-related issues. A list of design guidelines was derived by this first analysis. Once the most general guidelines are obtained, it is possible to analyse *Level 2* and *Level 3* to obtain more specific design guidelines. Indeed, *Level 3* contains information regarding the topology of components installed/assembled. At this level, specific guidelines were obtained referring mainly to the geometrical features of the elements to be installed. Guidelines such as, *keep weight below a given threshold P* or *avoid Module X in the working area A* were recorded.

Component	Score
Module P	1.00
Module L	0.95
Module D	0.87
Module Z	0.84
Module E	0.81
Module G	0.78
Module H	0.48
Module F	0.41
Module I	0.35
Module V	0.29
Module T	0.20
Module O	0.18
Module B	0.17
Module M	0.11
Module A	0.11
Module N	0.11
Module C	0.10
Module U	0.10
Module R	0.10
Module Q	0.08
Module S	0.03

Figure 9. Results analysis for the nose fuselage case study

By analysing each level (*Level 1*, *Level 2* and *Level 3*) it is possible to obtain guidelines referred to the domains present in the level of interest. For the above-mentioned case-study, design guidelines for the Interface Domain were derived from the *Level 1*, for the Ergonomic and Assembly Domains from the *Level 2*, while for the Component Domain from the *Level 3*.

4. Results

From the results analysis, the three highest modules addressed by the proposed approach are: Module P, Module L and Module D. To give an example about the use of the retrieved results for the identification of design guidelines oriented to assembly improvement, the Module P is described in detail analysing the interfaces connected to this module.

Looking at the *Level 1* of this framework, it is possible to notice that Module P is sharing 12 Interfaces with other modules of the product. Among these 12 Interfaces, the highest score among the four domains is the “ergonomic score” for Interfaces 1, 2, 9 and 10 (Figure 10). From this general representation, the following general guidelines (low level of granularity) were retrieved:

- Improve Interface 1 installation ergonomic features (i.e. change the interface route whenever possible or reduce the distance between Module P to the other Module connected by Interface 1)
- Improve Interface 2 installation ergonomic features (i.e. change the interface route whenever possible or reduce the distance between Module P to the other Module connected by Interface 2)
- Improve Interface 9 installation ergonomic features (i.e. change the interface route whenever possible or reduce the distance between Module P to the other Module connected by Interface 9)
- Improve Interface 10 installation ergonomic features (i.e. change the interface route whenever possible or reduce the distance between Module P to the other Module connected by Interface 10)

Main Level				
Module - Interface	Ergonomic Score	Assembly Score	Component Score	Interface Score
Module P - Interface 1	3.85	2.50	2.94	2.33
Module P - Interface 2	4.15	2.50	3.56	3.00
Module P - Interface 3	2.22	2.50	2.52	2.00
Module P - Interface 4	3.12	2.50	2.94	2.67
Module P - Interface 5	3.19	2.50	3.56	3.00
Module P - Interface 6	3.37	2.50	2.52	2.00
Module P - Interface 7	3.04	2.50	2.52	2.33
Module P - Interface 8	3.63	2.50	3.11	2.33
Module P - Interface 9	3.86	2.50	2.52	2.00
Module P - Interface 10	3.85	2.50	2.94	1.67
Module P - Interface 11	3.43	2.50	3.11	1.67
Module P - Interface 12	2.58	2.50	2.94	1.67

Figure 10. Results - main level

Other design guidelines can be defined in this phase having the same level of granularity. Moving into *Level 2*, attributes related to the ergonomic domain were analysed to identify more specific design guidelines. It is worth noting that the highest score is related to the attribute *Working Space Size* and *Zone* (Figure 11). From this assessment, the identifiable specific design rules for the ergonomic domain to reduce the assembly complexity are:

- Avoid assembling Interface 10 in Zone Lower
- Avoid using for Module P High Variety of Tools

Second Level									
Name	Working area	Zone	Access	Working space size	Variety of tools	Process	SCORE Zone	SCORE Access	SCORE Working Space Size
Module P - Interface 1	Working Area D	Lower	Access A	Big	Medium	easy	3	1	2
Module P - Interface 2	Working Area F	Lower	Access B	Normal	Medium	easy	3	1	3
Module P - Interface 9	Working Area F	Lower	Access B	Normal	Medium	easy	3	1	3
Module P - Interface 10	Working Area A	Upper	Access C	very small	Medium	easy	5	4	5

Figure 11. Results - second level

5. Conclusion

In the proposed work, a methodology to derive design guidelines at the *Conceptual Design* phase was explained. The method can be applied to obtain design guidelines considering a given target. This will help to reduce the solution space available at the *Conceptual Design* phase, limiting the almost infinite alternatives and guiding designers to definition of an improved product architecture. The proposed method is scalable, which means that it can be applied to several products and used to analyse different targets. It presents many advantages with respect to other knowledge gathering method: (i) it is applicable at the conceptual level, (ii) it considers attributes related to the optimization goal and, (iii) it takes into account the combination of attributes to understand in which way they affect the optimization target. The method presents the following drawbacks: (i) some of the design guidelines obtained might be already known to senior engineers, (ii) it requires a deep knowledge of the product in order to set up the hierarchical structure (i.e. attributes, domains, levels) and, (iii) it requires a large amount of data to fill the documents to perform the analysis. The method has been tested on a complex system providing an interesting case-study for the validation. Few of the derived design guidelines were already known by senior engineers but the method appeared to be a great support for the definition of new product architecture and helped with the understanding of some pain points and bottlenecks. Another interesting aspect in this design context is the assessment of the derived product architecture with numerical analysis. A method to assess product architecture is an on-going research as reported in the preliminary work of the same authors (Bouissiere et al., 2019). This method, together with the method presented in this paper, provides a robust methodology that can be used to converge to feasible product architecture solutions. Finally, the method presented is an on-going research. Indeed, during the research and development of new product architectures, designers has the possibility to get lost in the elevate number of alternatives. In addition, the application of design guidelines might generate negative repercussions elsewhere in the product architecture. To tackle this issue the approach requires the integration of enhanced mathematical tools (es. Artificial intelligent, etc.) able to spot the best alternative, driving designers in their choices. Future development will consist on: (i) automatize the data collection in order to minimize the time required to run the analysis, (ii) implement the derived design guidelines in a company-database to recall them when new products architectures are studied, (iii) implement the method inside the already existing CAD system to actively guide designers through the definition of the product architectures together with advance mathematical models.

References

- Boothroyd, G., Dewhurst, P. and Knight, W.A. (2010), *Product Design for Manufacture and Assembly*. s.l.:Taylor & Francis Group. <https://doi.org/10.1201/9781420089288>
- Bouissiere, F. et al. (2019), "Conceptual Design for Assembly in aerospace industry: A method to assess manufacturing and assembly aspects of product architectures", Delft, s.n. <https://doi.org/10.1017/dsi.2019.303>

- Dahmus, B.J., Gonzalez-Zugasti, J.P. and Otto, K.N. (2001) "Modular product architecture", *Design Studies*, Vol. 22 No 5, pp. 409-424. [https://doi.org/10.1016/s0142-694x\(01\)00004-7](https://doi.org/10.1016/s0142-694x(01)00004-7)
- Demoly, F. et al. (2011), "An assembly oriented design framework for product structure engineering and assembly sequence planning", *Robotics and Computer-Integrated Manufacturing*, pp. 33-46. <https://doi.org/10.1016/j.rcim.2010.05.010>
- Favi, C. et al. (2019), *Design for Assembly in the Conceptual Development of Aircraft Systems*, Modena, Springer, pp. 268-278. https://doi.org/10.1007/978-3-030-31154-4_23
- Favi, C. and Germani, M. (2012), "A method to optimize assemblability of industrial product in early design phase: from product architecture to assembly sequence", *International Journal on Interactive Design and Manufacturing*, Vol. VI No. 3, pp. 155-169. <https://doi.org/10.1007/s12008-012-0147-y>
- Fiorineschi, L. et al. (2015), *Product architecture definition: evaluating the potentiality of TRIZ tools*, s.l., Elsevier. <https://doi.org/10.1016/j.proeng.2015.12.415>
- Golob, B., Jezernik, A. and Hren, G. (2002), *A feature based approach for conceptual design*. Dubrovnik, s.n.
- Namouz, E., Summers, J.D. and Mocko, G.M. (2012), "Reasoning: Source of variability in the Boothroyd and Dewhurst assembly time estimation method", s.l., s.n. <https://doi.org/10.1115/detc2012-71075>
- Otto, K.N. and Wood, K.L. (2001), *Product Design: Techniques in Reverse Engineering and New Product Development*. s.l.:Prentice-Hall.
- Pahl, G. et al. (2007), *Engineering Design*. s.l.:Springer. <https://doi.org/10.1007/978-1-84628-319-2>
- Stone, R.B., McAdams, D.A. and Kayyalethekkel, V.J. (2004), "A product architecture-based conceptual DFA technique", *Design Studies*. <https://doi.org/10.1016/j.destud.2003.09.001>
- Ulrich, K. and Eppinger, S. (2011), *Product design and development*. V ed. s.l.: McGraw-Hill Education.