CDFA method: a way to assess assembly and installation performance of aircraft system architectures at the conceptual design

Giovanni Formentini\textsuperscript{1} \textsuperscript{1} · Francois Bouissiere\textsuperscript{2} · Claude Cuiller\textsuperscript{2} · Pierre-Eric Dereux\textsuperscript{2} · Claudio Favi\textsuperscript{1}

Received: 17 November 2020 / Revised: 23 July 2021 / Accepted: 14 November 2021 / Published online: 18 January 2022
© The Author(s), under exclusive licence to Springer-Verlag London Ltd., part of Springer Nature 2021

Abstract
This paper describes an engineering design methodology, called conceptual design for assembly (CDFA) in the context of aircraft development, to assess aircraft systems’ installation during conceptual phase, in relation to industrial performance objectives. The methodology is based on a given framework (hierarchical structure) which includes a set of attributes, collected in recognized domains that characterize the aircraft systems installation. The framework of the CDFA methodology enables to analyze product architectures at different levels of granularity, splitting the global analysis into sub-problems (problem discretization) with the aim to help architects and designers to identify product architecture weaknesses in terms of fit for assembly performances. The CDFA methodology was applied on a complex system (the nose-fuselage of a commercial aircraft) presenting a high number of criticalities both for the product and its assembly operations. Results identified the architectural components leading to the less efficient assembly operations and the rationales enabling to elaborate alternative architectures for an improved product industrial efficiency.

Keywords Product development · Architectural design · Design methodology · Conceptual design · Aircraft design · Design for manufacturing and assembly · Fit for assembly

1 Introduction

The global demand for commercial aircraft significantly increased in the past decades, and despite the recent COVID-19 pandemic situation, this trend is confirmed. In this context, aircraft manufacturers are called upon to improve the industrial performances keeping under control industrial investments (Maropoulos et al. 2014; Yancey 2016), safety requirements (certification) (Hickie 2006) and production lead time (Brusoni and Prencipe 2011). With this aim, product engineering and manufacturing engineering need to be integrated at the earliest design phase to design product architecture meeting both product and industrial performances.

Among all the design phases described in the product development process (PDP), the conceptual design phase is aiming at identifying the most suitable product architecture, providing the highest degrees of freedom. In this phase, any architectural change has no cost effect on the product until decision is made on the product architecture. In literature, many methods have been presented to consider the assembly aspects during the PDP; they are known as “Design for Assembly” (DFA) and/or design for manufacturing and assembly (DFMA). However, DFA and DFMA methods are mainly applicable at late design phase (i.e., embodiment design and detail design) when detailed information is available. The value brought by the development of DFA/DFMA methods at the conceptual design phase relies on the early consideration of the industrial dimension which is currently the main gap observed within the literature analysis.

The methodology described in this paper, called conceptual design for assembly (CDFA), translates product architecture data into a set of numerical values associated to fit for assembly assessment criteria (attributes), which are then integrated into a single value for a given assembly/manufacturing aspect (domain) and enabling product architectures comparison. The methodology is an applied research in the engineering design context starting from aeronautical industry needs and requirements. The term conceptual design for
assembly is already known and used within the engineering literature (Lombeyda and Regli 1999; Stone et al. 2004). However, concepts and frameworks presented in the mentioned works discuss the topic from a different research angle (i.e., software solution for a collaborative CAD-based design environment, or integration of conceptual and detail information for the assessment of assembly performance by using time-based method). In the present work, the CDFA methodology covers novel aspects in this field as follows: (i) it collects assembly/manufacturing knowledge switching from tacit implicit knowledge (unstructured information) to explicit knowledge (scoring matrices with numerical data), (ii) it avoids the use of detailed design information (i.e., the assembly time) in favour of design data available at the conceptual design phase, (iii) it creates a mathematical model and a framework able to consider the overall aircraft assembly problem by using sub-problems that are limited in terms of complexity (problem discretization) and (iv) it provides information to the designer regarding weaknesses of the product architecture in terms of assembly operations efficiency. In addition, by analysing the results of this methodology, it is possible to use the numerical values to elaborate architectural design guidelines. These guidelines can be used to derive optimized product architecture able to minimize the assembly and installation complexity. The optimization process usually starts from a given architecture and the context in which the CDFA methodology is used is the re-design phase rather than a novel design/idea. The use of functional model leads the system engineers/architects to change the way the module is built as well as its position, consequently a functional model is necessary to find alternative design solutions and alternative architectures. The CDFA methodology is considered a decision-making tool developed to assess the performance of an aircraft product architecture in relation to the assembly/installation phase. It is worth noting that, due to the specific context, usually alternative product architectures are developed within the same manufacturing/production framework due to many reasons as follows: (i) the high cost of changes related to the production sites, (ii) the adaptation and validation of test procedure and equipment and (iii) the reluctance of changes about consolidated activities and technologies which guarantee high standard in terms of safety and quality (one of the most important pillars to develop an aircraft). All these aspects contribute to the reactivity of the manufacturing/production sites to implement disruptive concepts and ideas even if the CDFA methodology does not prevent the possibility to assess high-innovative architectures.

The paper is structured as follows: after the literature analysis related to the topic of this paper (Sect. 2), the Conceptual Design for Assembly Methodology is presented (Sect. 3). The Case Study provides an application of the methodology on the nose-fuselage of a civil aircraft (Sect. 4). Then Results obtained by the application of the method are presented (Sect. 5). Finally, Conclusion discusses limitations and future developments of the presented approach (Sect. 6).

2 Literature background

The PDP is articulated in several phases in which different disciplines are collaborating. The main work on the formalization of the PDP is proposed by Pahl et al. (2007) where the following four (4) phases necessary to design a product are identified: (i) planning and task clarification, (ii) conceptual design, (iii) embodiment design and (iv) detail design. The PDP proposed by Pahl et al. (2007) considers only the design process itself without considering interactions between designers and other departments, such as the manufacturing department. Boothroyd et al. (2011) proposed a new PDP (nPDP) in which interactions among departments are considered since the initial phase of the design process. This new paradigm has led to the birth of the concurrent engineering (CE), whose purpose is the optimization of the design process by considering all product aspects from the beginning, breaking barriers between different departments (Lyu and Chang 2010). CE product development leads to many advantages such as reduced lead times (Tseng and Abdalla 2006), even if it requires a higher effort to manage the elevate number of information and stakeholders involved at the same time (Jun et al. 2006).

Several methods are counted as part of the nPDP, such as design for X (DFX) methods (Huang et al. 1999; Kuo et al. 2001; Holt and Barnes 2010), where the x is replaced with the optimization target (e.g., assembly, cost, maintenance, etc.). The DFMA is the first family of DFX methods that was developed. DFMA methods aim at optimizing the process and cost of manufacturing activities of subsystems by using the product own proprietary data (Coma et al. 2004). Cost and lead time are the main drivers associated with the manufacturing and assembly aspects. This is particularly true for big and complex products such as aircraft, where assembly and installation can impact over the 40% on the final cost (Bullen 1999). The most-used DFMA methods are the following: (i) the Boothroyd–Dewhurst (B&D; Boothroyd et al. 2011), (ii) the Hitachi method (AREM; Suzuki et al. 2003) and (iii) the Lucas method (Stone et al. 2003).

In the aerospace field, few attempts in the development of DFMA methods have been done. Lockett et al. (2014) applied DFA techniques with the help of CAD software to identify small changes to improve wiring system installation tasks. Barbosa and Carvalho (2014) collected a set of design guidelines applicable during the aircraft design to improve the assembly and manufacturing performance. Moreover, Butterfield et al. (2007) proposed a method to make use of
digital manufacturing techniques to evaluate the assembly process of an aircraft fuselage in the final assembly line. However, the adoption of DFMA methods is limited and few issues were highlighted within the literature. The first one is the elevate number of information required to use the derived DFMA methods that are mainly available at late design phases. The second one is the presence of constraints such as safety or weight limitations which are currently not addressed within the available DFMA methods, reducing the amount of potential architectural alternatives.

While state-of-art DFMA methods require detailed information available at the final stages of product design (El-Nounu et al. 2018), working at the conceptual level forces designers to use partial and high-level information such as product requirements, functions, etc. Starting from the basic approaches (i.e., functional and modular analysis), few attempts were done to adapt DFMA methods for the conceptual design phase providing as outcome the module analysis for assembly time reduction (Stone et al. 2004) as well as the assessment of the product layout with the assembly sequence (Favi and Germani 2012). Most of these works can be considered a baseline for the development of CDFA methods related to aerospace products but limitations related to the product complexity and required information details have been observed. Indeed, concerning the aerospace sector, only a limited number of works describing DFA methods and tools applicable at the conceptual level was observed. Among all, Domeshak et al. (1994) developed a tool that, by collecting previous aircraft design, can be interrogated to provide useful information during the conceptual design phase. Other works moved the research angle from the product to the final assembly line (Butterfield et al. 2007; Mas et al. 2013, 2016; Gómez et al. 2016). The optimization of the final assembly line is an important topic referring to the reduction of manufacturing and assembly costs, but it requires to be integrated with product design to be efficient.

The main challenge working at the conceptual phase is the availability of data which is characterized by high granularity. In this phase, the collected information is represented by schemes that encapsulate functions required by the product. There are many methods in literature to create schemes to represent the collected data. For instance, the product functional representation is one of those and it is characterized by black boxes connected with basic flows (i.e., energy, material and signal) (Pahl et al. 2007). Other tools and methods can be used to derive the scheme representing the product functions, such as the function means tree (Hubka and Eder 1989) and the functional evolution process (FEP; Shimomura et al. 1998). The arrangement of functional elements into physical chunks which become the building blocks for the product or family of products is known as “Product Architecture” (Ulrich 1995). There is no one-of-a-kind approach to represent product architecture.

The design structure matrix (DSM) is a powerful tool to describe interactions between product architecture components using a square-matrix (Browning 2016). However, the assessment of product architectures requires mathematical models that need to be validated in each context considering specific constraints. Concerning the optimization of product architecture, the creation of a modular product leads to advantages such as product differentiation (AlGeddawy and ElMaraghy 2013), parallelization in design and testing (Ethiraj and Levinthal 2004), flexible manufacturing and synergic cooperation with suppliers (Jiao et al. 2007) and creation of product family (Ma and Kim 2016; Bonvoisin et al. 2016; Ethiraj and Levinthal 2004). The improvement of a given product architecture usually requires the analysis of modularity. From the literature, it emerged that modular product architecture can positively impact the assembly phase (Stief et al. 2020). While the benefit of modularity is acknowledged for these technical aspects the over-increasing degrees of modularity in design decisions may lead to opposite results (i.e., increment of costs) (Engel and Reich 2015; Engel et al. 2017; Bonvoisin et al. 2016). The concept of modularity was introduced in the aeronautical sector (Monnoyer and Zuliani 2007) creating independent sub-parts or modules (Frigant and Talbot 2005; Erens and Verhulst 1997; Miller et al. 2002; Fricke and Schulz 2005; Siddiqi and de Weck 2008; Jung and Simpson 2017). However, the creation of fully modular aircraft is an on-going challenge (Atasoy et al. 2012).

The literature analysis highlighted how the conceptual design phase is the most critical phase to prevent installation and assembly issues. Aerospace industries are facing an integration challenge between product concept and manufacturing (i.e., assembly and installation) since product complexity is increasing, requiring a huge number of operations impacting cost and resources needs. Preliminary studies tried to tackle assembly and manufacturing aspects within the conceptual framework of product development but several limitations were observed such as the problem formalization for product architecture assessment, the mathematical models (framework) to link assembly parameters, as well as the quantitative assessment of assembly issues referring to information available at the conceptual design.

### 3 Conceptual design for assembly methodology

The proposed methodology, called CDFA (conceptual design for assembly) is divided into three main phases as follows: (i) product functional decomposition, (ii) architecture geometrical definition and (iii) Conceptual Design for Assembly assessment. Each phase is characterized by
different steps and design tools as reported in Fig. 1. The following paragraphs describe in detail each phase of the CDFA methodology.

3.1 Phase 1: product functional decomposition

The product functional decomposition is the starting point of the analysis. It enables to identify functional modules and their functional interconnections that will then be used to define the physical modules and their physical interconnections. Modules and their interconnections will then be characterised to perform the assembly assessment. Product functional decomposition is of a great importance in this field which is characterized by consolidated design solutions and technologies. While keeping the compliance with stringent requirements for this type of product, the design of new product architecture at the conceptual design phase (e.g., module layout/arrangement, module position, module integration/decoupling, module assembly/installation, module fixation, interface routing, interface installation, etc.) is allowed. For this reason, this initial phase, even if limited to initial choices about technology and design solution, still provides an important window on how possible changes can be developed at the conceptual level. The product functional decomposition is performed in two steps: (i) functional derivation, and (ii) module derivation. Literature approaches have been used

---

Fig. 1 Conceptual design for assembly methodology
with some adjustments (i.e., customization) to proceed with other methodology steps.

3.1.1 Functional derivation

The functional derivation applies the functional analysis (Pahl et al. 2007) to the product under study to obtain the product functions. The functional analysis consists of defining functions and sub-functions (main and auxiliary) through a hierarchical scheme and basic fluxes that link these functions (Kroll 2013). Basic fluxes identified by Pahl et al. (2007) are material, energy and signal. In the proposed methodology a modified version of the functional analysis proposed by Pahl et al. (2007) is used to characterize the given product (aircraft) with the support of a dedicated tool. Indeed, within the same functional flow (e.g., material flow) different types of fluxes are determined and associated to a given colour (see Table 1). For instance, assuming that two generic functions are connected by a flux of material, it is possible to further specify the type of material (e.g., gas) and the sub-type (e.g., air). A given colour is assigned to the type of material (e.g., green for gas type) and a unique RGB code is assigned to the sub-type (e.g., 7, 255, 62 for the air). The outcome of the modified approach is a graph presenting as many colours as different fluxes available for the whole functions. Moreover, the colour assignment is important to cover the issue related to people with colour vision deficiency (CVD) (Nuñez et al. 2018). Indeed, a general colour was assigned for each type of flow, then a gradation of the general colours is proposed for sub-flows to provide a CVD-safe colour map. Once colours are assigned, they must not be changed to avoid inconsistencies during the application of the CDFA approach on other case-studies. The presented functional derivation enables users to (i) improve readability of the functional representation, (ii) increase the level of detail of the functional analysis without requiring data from a lower design phase, (iii) provide better understanding of the implication of each requirement and (iv) facilitate the switch from fluxes (functional representation) to physical interconnections (physical representation). Table 1 shows all basic fluxes identified within an aircraft. These fluxes were defined by the authors following the previous work of Hirtz et al. (2002). These fluxes are then used to derive modules and subsequently their physical interconnections to create modules.

The functional decomposition provides potential benefits in terms of product abstraction, functions characterization and fluxes identification. However, the process is laborious, time-consuming and requires collaborative sessions to be developed. Some functions can only be analysed in the context of a particular solution, limiting the design space as well as the ability of designers to think in abstract terms (Kroll 2013). Moreover, when the level of details in functional decomposition is too deep, it may lead to a lack of freedom for the designer adversely affecting innovation and creative performance (Leenders et al. 2007). Due to the specificity of the aeronautical field and aircraft product development, which is mainly characterized by very stringent regulations about safety requirements, and high cost of changes, the conceptualization of new ideas, carried out through a novel design process, requires a long time to be formalized, discussed, accepted, validated and tested. Within this framework, the criticisms highlighted for the method proposed by Pahl et al. (2007), although partially restrict the potential of this method, do not negatively affect the development of the CDFA methodology which focuses on the assessment of product architectures in terms of manufacturing and assembly. Besides, functional representation is a powerful tool to develop new module concepts and proceed towards the design of new architectures.

3.1.2 Module derivation

Once the functional scheme is obtained, it is necessary to derive modules. The derivation of modules requires first to move from functions to interfaces. Interfaces represent how functions physically interact with the system of interest. Interfaces are derived considering the basic flows that interact with the system and their type. The colour is inherited by the basic flux colour type (i.e., general colour), while, at this stage, the arrow’s type becomes meaningless. Thus, for the presented CDFA methodology, all interfaces are represented with solid arrows. It is worth noting that, while the functional representation identifies all the interactions that are present in the system of interest, some of them might not need an interface to connect with the system. For instance, the cabin crew interacts with the system to store luggage within the cabin (usually called hat-rack), and, in this case, the interface necessary to connect these two modules is the human interface. Since the purpose of the CDFA methodology is to analyse product assembly and system installation, the human interface is meaningless because the action to store luggage within the cabin is performed by the human. Following this principle, for aircraft systems only four interfaces are derived starting from the identified fluxes: (i) electrical, (ii) mechanical, (iii) fluid and (iv) air. Table 2 reports the four interfaces considered for the aircraft systems and the related matching with the basic flow types.

Once interfaces are defined, it is possible to proceed with the creation of modules. Modularisation aims at clustering the functions into modules (product decomposition into building blocks) with specified interfaces (Ericsson and Erixon 1999). Module derivation from functional analysis is a key step at the conceptual design stage, whether novel design or redesign. It provides an engineering view of how the sub-functions work together to achieve the desired
### Table 1: Basic fluxes for aircraft (extension of the basic fluxes provided by Hirtz et al. 2002)

| Aircraft Basic Fluxes | Sub-Type | Type Colour | Colour | RGB |
|-----------------------|----------|-------------|--------|-----|
| **Human**             | N.A.     | Black       | 0 0 0  | 0 0 0 |
| **Solid**             | Waste    | Brown       | 128 63 | 255 0 0 |
|                       | N.A.     | Black       | 0 0 0  | 0 0 0 |
| **Liquid**            | Fuel     | Blue        | 255 144 | 255 0 0 |
|                       | Oil      | Blue        | 124 252 | 0 0 0 |
|                       | Water    | Blue        | 125 252 | 0 0 0 |
| **Gas**               | Air      | Green       | 124 252 | 0 0 0 |
| **Energy**            | N.A.     | Black       | 0 0 0  | 0 0 0 |
| **Acoustic**          | Energy generated by turboprop motion converted into noise | Orange | 255 127 | 0 0 0 |
|                       | N.A.     | Black       | 0 0 0  | 0 0 0 |
| **Chemical**          | Energy produced by aircraft batteries | Green | 124 252 | 0 0 0 |
|                       | N.A.     | Black       | 0 0 0  | 0 0 0 |
| **Electrical**        | Energy transmitted by aircraft harnesses | Yellow | 255 255 | 0 0 0 |
|                       | N.A.     | Black       | 0 0 0  | 0 0 0 |
| **Hydraulic**         | Energy used to actuate the landing gear | Blue | 167 255 | 0 0 0 |
|                       | N.A.     | Black       | 0 0 0  | 0 0 0 |
| **Mechanical**        | Energy generated by a mechanical connection to fix the seat on the cabin | Purple | 138 255 | 0 0 0 |
|                       | N.A.     | Black       | 0 0 0  | 0 0 0 |
| **Pneumatic**         | Energy used by the pneumatic aircraft system | Brown | 165 255 | 0 0 0 |
|                       | N.A.     | Black       | 0 0 0  | 0 0 0 |
| **Thermal**           | Energy exchanged in the aircraft cooling system | Red | 255 255 | 0 0 0 |
|                       | N.A.     | Black       | 0 0 0  | 0 0 0 |
| **Control**           | Pilot that regulates air in the cabin | Sound | 250 128 | 0 0 0 |
|                       | N.A.     | Black       | 0 0 0  | 0 0 0 |
| **Feedback**          | Indicator of the fuel level in the aircraft | Sound | 255 255 | 0 0 0 |
|                       | N.A.     | Black       | 0 0 0  | 0 0 0 |
| **Signal**            | N.A.     | Black       | 0 0 0  | 0 0 0 |

**Explanation**
- **Human**: All or part of a person who physically interacts with the system.
- **Solid**: Object with mass and shape which physically interacts with the system.
- **Liquid**: Fluid which physically interacts with the system.
- **Gas**: Gas that physically interacts with the system.

**Material**
- **Solid**: Object with mass and shape which physically interacts with the system.
- **Liquid**: Fluid which physically interacts with the system.
- **Gas**: Gas that physically interacts with the system.

**Energy**
- **Acoustic**: Work performed to produce and transmit sound.
- **Chemical**: Work resulting from chemical reactions.
- **Electrical**: Work resulting from motion of electrons.
- **Hydraulic**: Work performed by moving fluids.
- **Mechanical**: Work performed by a mechanical system.
- **Pneumatic**: Work resulting from the motion of gas.
- **Thermal**: Work resulting from a thermal system.

**Control**
- Command sent to an apparatus to regulate it.

**Signal**
- Information about the state of the system.
functional requirements, independently of how the function is performed. The engineering definition of modules in aircraft systems presents many concerns related to the huge number of constraints that need to be satisfied (the presence of redundant elements placed in different areas for safety reasons). The method proposed by Stone et al. (2004) is adopted within this methodology with the aim to consider all the constraints required for the development of modules. This method is based on three heuristics (dominant flow, branching flow and transmission/conversion) and it allows to identify product modules by grouping sub-functions together. The list of modules retrieved by the adoption of this method can be used for concept generation and module instantiation. Among the different methods developed for module derivation (i.e., module heuristics, DSM and modular function deployment—MFD), the module heuristics is the most suitable for the scope of this work, since it shows important features which fit with the type of product under analysis and the level of confidence required to develop the modules in such a product. Module heuristics can capture flows describing the underlying physics of the product and it is more flexible than other methods that require a matrix or mathematical description (Borjesson 2010). The method presents a high repeatability on a given function structure diagram which is the result of the previous phase (functional derivation) as demonstrated by Hölttä-Otto (2005). The module heuristics is well supported by empirical research on hundreds of real products; however, it requires an engineering judgement (theoretical foundation is less scientific than the other methods) to be operated and it does not guarantee that the identified modules represent the optimum clusters. Nevertheless, it provides a consistent and.repeatable product structure breakdown. The application of the module heuristics suffers of software integration/implementation which would be beneficial for large projects as in the case of aircraft. This drawback is counterbalanced by flexibility, indeed it is worth nothing that, to increase the level of confidence in the definition of suitable modules in such a critical product, a mapping with existing modules for a given product is possible. In this case, the module heuristics method (top-down approach) is coupled with the analysis of available product structure (bottom-up approach) with the aim to match the existing modules with the ones retrieved by module heuristics. If the goal of the CDFA assessment is the optimization of a given architecture, this task is necessary and it will lead to the following two results: (i) keep the level of confidence about modules derivation in relation to aircraft products that are characterized by many design constraints and (ii) identify possible alternative solutions for modules definition (i.e., module splitting/merge). On the other hand, if the goal is to assess a new concept (i.e., concepts that are newly developed), module breakdown is fully based on module heuristics (top-down approach) increasing the design solution space but downgrading the level of confidence in module definition. A one-to-one mapping between modules and functions is the easiest solution to consider aspects such as safety or operability requirements. However, this option can bring to product architectures with more modules (more options and a higher level of modularity), giving importance to the role of component interfaces increasing the difficulty of achieving product assemblability (Engel et al. 2017). Hence, having more modules is not always the right way to proceed in this phase and the use of module heuristics allow an engineering assessment of feasible modules based on initial requirements and given design decisions (e.g., combustion of fuel for power generation). Once modules are defined, it is possible to create the product physical architecture by linking modules with interfaces derived in the previous step (Fig. 2).

### 3.2 Architecture geometrical definition

The architecture geometrical definition consists in reading the data available in the derived product and translating them into numerical format. This phase is composed of three (3) steps: (i) Definition of geometrical product architecture, (ii) Definition of Conceptual Bill of
Materials, and (iii) Definition of knowledge scoring matrices (knowledge-based).

### 3.2.1 Definition of geometrical product architecture

The fit for assembly assessment at the conceptual level requires the conversion of conceptual features into parameters that can be visualized and measured. The information derived in the previous steps (i.e., functional modules and functional schemes) are used to create a virtual representation of the product under study. The use of a virtual environment (i.e., CAD tool) enables to represent data available at the conceptual phase into elementary geometries. The simplified digital mock up (sDMU) is a graphical and geometrical representation of a specific product architecture showing modules shape and interfaces among modules in a three-dimensional space. The sDMU is composed of 3D geometrical items such as boxes, cylinders, etc. that describes how modules and interfaces are built and interconnected (Fig. 3). The sDMU provides a visual representation of the product architecture obtained in the previous steps, enriching the product architecture itself with a new set of information. It presents more detailed information such as the distribution of modules inside the system of interest (module position), the overall module shape and volume (module bounding box) and many others. Information represented in the sDMU is enclosed in the functional scheme, for example by knowing how many functions and the type of functions collected in a functional module then module position, module bounding box, length of interfaces, etc. can be estimated to create the sDMU. The level of detail and the granularity of the sDMU evolves during the application of the methodology. If the methodology is used to analyse new products, then the sDMU starts with a low-level of detail (i.e., low granularity) and it is enriched with more detailed information when later design phases are approached. On the other hand, if the methodology is based on an existing product of which a CAD file is available, it is possible to simplify the CAD file by neglecting all details and representing modules as boxes with their bounding box and interfaces as cylinders whose diameters reflect the real dimension.

The sDMU is providing the minimum set of data enabling to perform the fit for assembly assessment. The

| sDMU item | Information collected in sDMU |
|-----------|--------------------------------|
| Module    | Position \((x, y, z)\) based on a given reference |
|           | Shape (i.e., rectangular box, cylinder) |
|           | Bounding box |
|           | Colour (i.e., blue) |
| Interface | Position \((x, y, z)\) based on a module in/out reference |
|           | Path |
|           | Overall length |
|           | Shape (i.e., cylinder) |
|           | Size (i.e., diameter) |
|           | Colour (based on interface type) |
required set of data necessary to build a sDMU is reported in (Table 3).

### 3.2.2 Definition of conceptual bill of materials

The conceptual bill of materials (cBoMs) is a document aiming at capturing the functional and geometrical data of the previous steps to enable computation of the fit for assembly scoring. The cBoM presents a table-form in which each row represents an interface between modules and each column represents an attribute that is characterizing a particular interface. The cBoM is characterized by a hierarchical structure, subdivided into levels (layers), domains and attributes. The hierarchical structure is the methodology framework that allows to combine attributes for a given problem (i.e., the overall assembly or a sub-assembly). The cBoM structure enables a decomposition of the given problem in sub-problems allowing to incorporate aspects from different level of granularity that otherwise might be discharged.

The proposed framework is a description of the product enabling to identify the impact of each attribute on the assembly process. The cBoM is collecting product data through fixed information which characterizes a specific level. Information must be enclosed in the fixed information part are collected in Table 4.

If necessary, other information can be added within this framework, depending on the level (layers) and the type of product under study.

#### 3.2.2.1 Definition of levels (layers)

A level is defined as a group of data which is modelling the main feature characterized a specific sub-problem of the overall product assembly. Different levels can be defined according to the available information. To switch from a level to another one is necessary to identify the product invariant. A product invariant is a design feature that does not change and cannot be changed within the product under study. The definition of the invariant allows to specify information with respect to it. For each defined level, the following actions are necessary: (i) to identify possible invariants that link two neighbouring levels, and (ii) to express the relation that exists between two neighbouring levels using invariants.

For example, if the "space distribution" (i.e., product areas) in a product is fixed and cannot be changed; then the “space distribution” can be considered as an invariant. The identified invariant allows to split the global analysis into sub-problems that are limited in terms of complexity (problem discretization).

#### 3.2.2.2 Attributes and domain definition

Attributes and domains that characterize the main criteria of fit for assembly performances are defined on a knowledge basis, with a CE approach (i.e., involvement of manufacturing department, architecture designers, operators, etc.). A mathematical model is then created to operate on these criteria providing the fit for assembly assessment.

#### 3.2.2.3 Attributes class definition

An attribute (A) is a key feature that influences assembly operations. Giving a generic attribute A [1 to t; x ∈ N>0], the attribute A describes a specific aspect related to the assembly operation. To define an attribute, it is necessary to indicate the following: (i) the name of the attribute, (ii) the dimension (i.e., the unit measure or the quantity) and (iii) the level in which it is available.

The definition of an attribute can be set after a deep study of the product under development. A list of key attributes reflecting the assembly complexity can be obtained knowing the design phase in which the methodology is applied (e.g., conceptual phase, detail phase, etc.). Attributes might be of interest or not, according to the level in which they are placed.

#### 3.2.2.4 Domain class definition

A domain (D) is a cluster of one or more attributes (A) that address the same assembly aspect and provides the same meaning for each designer/engineer. Giving a generic domain D [1 to t; t ≤ x; t ∈ N>0], the domain D is a vector characterized by n attributes [n ≥ 1; n ∈ N>0]. For each level of the cBoM framework, domains

| Table 4 | Fixed Information for the cBoM document |
|---------|----------------------------------------|
| **Title** | **Type** | **Description** |
| Name     | string     | It is the name associated to the element under study. It can be chosen arbitrarily by the user |
| ID       | integer    | It describes the ID of the element under study. It can be generated according to a specific rule, or it can be chosen arbitrarily by the user |
| Interface type | string     | It identifies the type of interface; it is compliant with interfaces identified in the functional scheme. It can be defined by the user following the interface type reported in Table 2 |
| Module IN | string/integer | It represents the module where the interface starts |
| Module OUT | string/integer | It represents the module where the interface ends |
can have a different number of attributes. By clustering attributes into domains, and placing domains into levels, the hierarchical structure is obtained (Fig. 4).

3.2.3 Definition of knowledge scoring matrices

The formalization of the knowledge is a great challenge and the research field associated to it is called knowledge engineering (KE). Relevant knowledge might appear in different forms (e.g., technical drawings, spreadsheets, etc.) and in unstructured manners but the main challenge is how to deal with it in terms of collections, formalization and utilization (Staab et al. 2001; Ahmed 2005; Reed et al. 2011). The approach proposed within this methodology to formalize the knowledge focuses on the following two aspects: knowledge acquisition and ontology definition (Guarino 1995). The knowledge acquisition relies on a knowledge-based concurrent approach (Favi et al. 2019). The method enables to retrieve the knowledge through the definition of scoring matrices. The structure and the vocabulary (i.e., ontology) of the knowledge must follow three principles: (i) role-limiting, (ii) knowledge typing and (iii) reusability.
The ontology proposed within this methodology is based on the definition of knowledge scoring matrix (kSM). The kSM is a table of \( Y \) rows, where each row corresponds to a score (i.e., from 1 to \( m \)). The kSM form is different according to the information it processes. For example, if the kSM translates strings information (i.e., \( Y \in \Sigma \) with \( \Sigma \) be the set of alpha-numerical strings without the null element), then to each string a score is associated (Fig. 5: \( sx \)). On the other hand, if the kSM translates numerical data (i.e., \( Y \in \mathbb{R} \) with \( \mathbb{R} \) be the set of real numbers), then to a given range of values a score is associated (Fig. 5: \( dx \)). By following this approach, a scale from 1 to 5 (Likert scale) has been used, since considered the most suitable for the specific problem.

The kSM is necessary to translate the collected information into dimensionless values and perform mathematical computations. The kSM shall be created for each attribute in the cBoM and is composed of the following: (i) name of the attribute, (ii) numerical range or string and (iii) score. It is important to notice the maximum score available for all the kSM must be the same. A good practice during the creation of knowledge scoring matrices is to express the rationale behind each score and to annotate it together with the kSM. In fact, the kSM represents the current industrial capabilities associated to the product technological level. The kSMs are updated every time an improvement of the assembly and/or manufacturing plant is performed.

### 3.3 Conceptual design for assembly assessment

The architectural analysis is divided into the following two phases: (i) normalization process, and (ii) architectural assessment.

#### 3.3.1 Normalization process

The normalization process consists in transforming the data inside each attribute into dimensionless value switching from heterogeneous data (i.e., string, number, etc.) to homogeneous data (i.e., dimensionless scores) to perform mathematical operations. The Normalization process requires two inputs: (i) the knowledge scoring matrices, and (ii) data for each attribute. For example, if the interface 1 presents a generic value for the attribute “length” (i.e., 3.5 m) then the normalization process will translate the value into a dimensionless score according to the kSM associated to the attribute “length” (i.e., 1). Once the normalization process is completed for all attributes, all data in the cBoM are normalized and dimensionless, enabling to proceed further with the architecture assessment. The normalization process can be performed by using a spreadsheet to avoid a prone-to-error task.

#### 3.3.2 Architectural assessment

The product architecture is analysed by using the cBoM framework. The analysis starts from the lowest level (i.e., Level \( l \) reported in Fig. 6), by considering the first ID (i.e., \( i_1.l.1 \) where \( l \) indicates the digits representing Level \( l \)). The function \( g(\bullet) \) is a mathematical operator which is applied inside each domain to collect scores of each attribute associated to the same ID. The choice of mathematical operator used to collect scores is addressed later within section “Mathematical Operators”.

This process is repeated for each ID. The output of the first computation (Level \( l \)) is a single score for a given ID representing the information for a specific domain. When all IDs of level \( l \) have a single score associated for each domain (i.e., domain SCORE), then the IDs which have the same first digits are collected with the function \( f(\bullet) \) which is a mathematical operator (e.g., \( i_1.l-1.1, i_1.l-1.2 \), etc.), defined according to the level invariant. The output of function \( f(\bullet) \) is a single score for interface that has the ID of the upper level. Indeed, the domain of the upper level (i.e., Level \( l-1 \)) is automatically filled by the lower level (i.e., Level \( l \)). The

![Fig. 5 kSM definition for string (sx) and numeric data (dx)](image_url)
output of this process is represented in Fig. 6. The process is repeated for all levels, clustering the results of all interfaces, with the aim to obtain a single score for each domain. Interfaces and scores are collected in the final domain called “Main Level” which presents scores for all domains defined in the cBoM and all interfaces of the product under study.

The aim of the architecture assessment is to obtain a single score for each Interface that characterises the assembly complexity of all product interfaces. Thus, it is required to further collect scores inside the Main Level with the mathematical operator $h(\cdot)$. Once the function $h(\cdot)$ is applied, one single score for each Interface is obtained (Fig. 7). This is the final results of the CDFA assessment.

The scores describe the interfaces assembly complexity considering all the parameters involved in the assembly and installation processes. Scores can be sorted out from lowest to highest to identify elements (interfaces and modules) that are the most complex to install inside the aircraft and that required to be redesigned.

### 3.3.2.1 Mathematical operators

The process of translating data from one level to another one is achieved through mathematical operators. A mathematical operator is a function—$y = f(x_1, x_2, \ldots, x_n)$—that combines inputs ($x_1, x_2, \ldots, x_n$) to obtain one single output ($y$). For the proposed methodology, the mathematical operators available in literature can be divided into the following two groups: weighted operators and weight-less operators. The former enables to apply weight on the input data. In this group there are multi-attribute-decision-making (MADM) methods (i.e., technique for order of preference by similarity to ideal solution—TOPSIS) (Hwang and Yoon 1981). On the other hand, the mean operator, the root mean square (RMS) and the average square are some of the available weight-less operators. The choice of the mathematical operator is driven by the following: (i) the invariant chosen in the analysis, (ii) the uncertainty whose is affecting the input data and (iii) the importance given to outliner/inliner data. The mean operator is used to collect data when the values of the distribution do not differ too much from each other. For instance domains’ scores are gathered
using the mean operator since, through the data normalization and score collection, domains’ scores do not contain discrepancy. On the other hand, the RMS is used to collect scores when data might have discrepancy due to: (i) errors in the data collection process, (ii) the creation of clusters of information which are not strictly related (e.g., total length of harness and number of connections belong to the same domain, but these attributes are not directly affecting each other). Thus, RMS is usually adopted to cluster scores of different attributes within a domain.

4 Case study

The proposed methodology was applied to assess the fit for assembly performances of one of the most challenging system of a civil aircraft: the nose-fuselage (Fig. 8). Within the nose-fuselage many modules and connections are present (e.g., electrical cables, air ducts, hydraulic pipes, etc.) and they need to be installed and assembled within very confined areas. Moreover, the nose-fuselage architecture is constrained by several aspects, for instance the need to install elements inside an already-given structure (i.e., aircraft skeleton), the requirements to have elements redundancy for safety purposes, etc.

The goal of the presented case study is to numerically assess the nose-fuselage architecture with the aim to identify interfaces considered critical for the installation process, providing a list of the most critical items to install. The ranking can be considered as a very helpful tool to guide the redesign phase, providing hot-spots and rationales to identify and subsequently avoid critical assembly tasks. This information will then be used for design improvements which is outside the scope of this work.

4.1 Nose-fuselage functional decomposition

The nose-fuselage functional decomposition is composed of the following: (1) functional derivation, and (2) module derivation. The functional analysis was performed to derive all the nose-fuselage functions and interfaces among them. Functions were derived according to Pahl et al. (2007) using information available in literature (e.g., equipment functions, connection among equipment, etc.). The functional scheme was obtained considering a civil aircraft already available from the market. In fact, it is necessary to start from design actions already available to be able to derive a full functional scheme. Functions were derived using several documents including technical documents describing systems used in aircraft, technical drawing of the nose-fuselage, use and maintenance manual, accident and malfunctions reports, etc. The analysis of these documents allowed to obtain information regarding systems inside the aircraft, how they interact and how they are subdivided. The general function “Fly, manage flight and allow passengers entrance and carry” was considered, and then it was subdivided into more specific functions with a hierarchical structure. Four hierarchical levels were obtained. Figure 9 shows an extract of the second level (the overall functional scheme is not provided due to space limitation and confidentiality).

Once the functional scheme was obtained, a top-down approach based on module heuristics (Stone et al. 2004) was coupled with a bottom-up approach to derive modules. From basic fluxes identified in the functional scheme, it was necessary to switch to interfaces. To reduce the overall analysis perimeter, only electrical and air interfaces were considered within this work since from engineering perspective they were considered the most complex to install (Hermansson et al. 2013). In this specific case, since it is based on a real
product and the scope of the analysis is the evaluation of a given product architecture through the mean of the CDFA, the module heuristics was coupled with the analysis of available product structure (overlapping existing modules with the ones retrieved by module heuristics). It is worth noticing that in the product architecture development of aircraft systems, the optimization phase is mainly characterized by the module re-arrangement which means the possibility to split/merge modules, module reallocation, interface routing and re-architecture of interconnection among modules. Thus, the list of modules is mainly driven by the available solutions and modules derivation using module heuristics is an exercise done to investigate possible alternatives (split/merge modules) for module definition based on functional decomposition. An extract of the derived modules is shown in Fig. 10.

4.2 Nose-fuselage architecture geometrical definition

The sDMU was created following the conceptual modules, and architectural data were created making use of existing product definition information, in particular the following: (i) the bounding box of each module, (ii) the length of connections (interfaces) among modules, (iii) the modules and connections position and (iv) the connections diameters. Rectangular boxes were used to model modules while cylinders for interfaces. Interface colours are based on Table 2. For the case study in exam, since 3D models were already available, with the help of virtual reality technologies it was possible to draw interfaces, module connections and bounding boxes easily and in a straight-forward manner. The sDMU obtained is presented in Fig. 11.

Then the cBoMs were derived. For the product under study, it is composed of three levels (layers) of data as follows: Levels 1, 2 and 3. Level 1 is composed by only one domain called “Interface Domain” which includes attributes referring to the overall interfaces among modules. To switch from Level 1 to Level 2, the invariant working area was defined. Working areas are pre-defined areas identifiable in the nose-fuselage. These areas cannot be changed due to structural reasons (i.e., design of beams, skin and floor). The invariant working area is used also to
define attributes’ scores inside knowledge scoring matrices. Each attribute has a different score according to the working area in which is defined (Table 5). For instance, in the working area “Right Bay” the attribute “Zone” for the value “Middle” has a score of 4, while for the working area “Cockpit” the same attribute with the same value (i.e., Middle) has a score of 1.

Level 2 is composed by two domains called “Ergonomic Domain” and “Assembly Domain”. The first one includes four attributes referring to ergonomic aspects of the installation process, while the second one presents two
attributes representing complexity of the installation process itself (see Table 7).

To switch from the Level 2 to the Level 3 the invariant *interface* was defined. Interfaces are defined from the functional analysis and cannot be changed without changing the product. The invariant *interface* was used to define attribute scores inside knowledge scoring matrices (Table 6).

### Table 5 Extract of kSM for attribute “Zone” (II level)

| Invariant (working area) | Zone | Score |
|-------------------------|------|-------|
| Right bay Upper         |      | 5     |
| Right bay Middle        |      | 4     |
| Right bay Lower         |      | 4     |
| Cockpit Upper           |      | 3     |
| Cockpit Middle          |      | 1     |
| Cockpit Lower           |      | 3     |

The Level 3 presents one domain called “Component Domain” which includes attributes referring to physical elements composing the interface. All attributes and domains used for the analysis are summarized in Table 7.

After the definition of the cBoM structure, the *knowledge* scoring matrices for each attribute were defined according to the ontology developed. The following four meetings were organized: (i) an initial in person meeting, (ii) two follow-up *web* meetings and (iii) a final in person review meeting. Industry departments involved in the meetings are listed in Table 8.

During the initial meeting (first meeting), the methodology was presented. Then, in the first *web* meeting (second meeting) a survey was submitted to the manufacturing department to collect expertise from assembly operations and related tasks. Then, results of the survey were analysed to obtain the first draft of the kSMs. The second *web* meeting (third meeting) was organized with the Product Architecture and Design Department to show the kSMs obtained and few modifications were suggested. In the final meeting, the latest version of kSMs was presented and finalized. An extract of the kSMs obtained is presented in Fig. 12.

### Table 6 Extract of kSM for attribute “Length” (III level)

| Invariant (interface) | Length | Score |
|-----------------------|--------|-------|
| Air                   | 0 < x ≤ 2 | 1     |
| Air                   | 2 < x < 5 | 2     |
| Electrical            | 0 < x ≤ 5 | 1     |
| Electrical            | 5 < x < 10 | 2     |

4.3 Conceptual design for assembly assessment

To perform the final step, which is the Conceptual Design For Assembly assessment, it was necessary to (i) normalize all information collected in the cBoM using the derived *knowledge* scoring matrices, and (ii) provide the mathematical algorithm to collect the normalized data inside the cBoM to obtain a final global score for each interface. The mathematical algorithm was defined using three different operators: (i) RMS, (ii) mean operator and (iii) TOPSIS method. The RMS was chosen to collect attributes’ scores for each domain. The reason lies in the need to consider possible errors; in fact initial data present different roots (i.e., some data are measured, others are derived by engineering knowledge) and they might present some evaluation errors. The use of RMS allowed to obtain a more conservative result with respect to other mathematical operators.

The mean operator was chosen to collect score at each level (i.e., from Level 3 to 2, from Level 2 to 1). In fact, data are initially collected in the cBoM; then, using the *knowledge* scoring matrices, they are normalized (i.e., data are translated into scores) and collected per domain. After the normalization process and the domains’ collection, scores have all the same roots and no further source of error needs to be considered. Finally, the TOPSIS was used to obtained one single score (i.e., final global score) for each interface from domain scores. The TOPSIS method was chosen since it allows to apply weights on each domain to tune the overall assessment. In fact, due to the nature of the methodology itself, in the model definition some attributes and domains might be underestimated or overestimated in terms of assembly complexity. This may lead to some shortfalls that can be recovered afterward making use of weights. Indeed, weights can be used to increase/decrease the importance on the final global score of each domain. However, for the specific case-study, no weights were added since results showed to be in line with engineering judgment. The overall cBoM, where operators are stated explicitly is shown Fig. 13, while Fig. 14 shows an extract of the result of the CDFA assessment phase, where the five most critical interfaces (i.e., most complex to install) are displayed and highlighted in the sDMU.

5 Results

CDFA results provide an estimation of the complexity to install connections (interfaces) and associated modules inside the nose-fuselage. Final global scores range from 0 to 1 due to the application of the TOPSIS method. Interfaces with highest score represent the most complex to install. As general outcome, results show that electrical
interfaces are the most critical for the installation process. The hierarchical structure (framework) of the CDFA methodology enables to understand where criticalities lie. An example is reported below for the interface E45 which has the highest score among all interfaces as reported in Fig. 14. Indeed, Interface E45 connects Module N to S and presents an overall score of 0.83 and it is the most critical to install. Moving to the global scoring, a more accurate

![Table 7 Attributes and domains identified for the nose-fuselage](image)

| Domain                  | Interface type | Attribute               | Explanation                                                                 |
|------------------------|----------------|-------------------------|-----------------------------------------------------------------------------|
| Interface domain       | Air            | Total length of ducts   | Overall air interface length                                                |
|                        | Air            | Branches                | Number of times air interface branches out                                  |
|                        | Elect          | Total length of harness| Overall electrical interface length                                          |
|                        | Elect          | Number of connections  | Number of connections electrical interface collects                         |
|                        | Elect          | Number of straight nodes| Number of times electrical interface branches out                           |
| Ergonomic domain       | Air/elect      | Access                  | Access used to bring the interface inside the working area                  |
|                        | Air/elect      | Zone                    | Zone in which interface is installed                                         |
|                        | Air/elect      | Working space size      | Available space during the installation operations                          |
| Assembly domain        | Air/elect      | Variety of tools        | Number of tools necessary to perform the assembly                           |
|                        | Air/elect      | Process                 | Complexity of the installation process                                       |
| Component domain       | Air            | Air bends               | Number of air ducts elbow                                                    |
|                        | Air            | Air shape               | Shape of the air duct                                                        |
|                        | Air            | Air weight              | Weight of the air duct                                                       |
|                        | Air            | Air piece length        | Length of the air duct                                                       |
|                        | Elect          | Electrical weight       | Weight of the electrical cable                                               |
|                        | Elect          | Electrical piece length | Length of the electrical cable                                               |
|                        | Air/elect      | Fragility               | Breakability of the duct/cable material                                      |

![Table 8 Meetings participants](image)

| Department                        | Participants                      | Meeting |
|-----------------------------------|-----------------------------------|---------|
| Product Architecture and Design   | Aircraft Architect               | X       |
| Product Architecture and Design   | DMU operators                     | X       |
| Product Architecture and Design   | System installation designers     | X       |
| Manufacturing and Assembly        | Industrial architect              | X       |
| Manufacturing and Assembly        | Industrial routing designers       | X       |
| Manufacturing and Assembly        | Manufacturing operators (blue-collar)| X       |
| Manufacturing and Assembly        | Ergonomic expert                  | X       |

![Fig. 12 Derived knowledge scoring matrix for two different attributes (total length of harness and working space size)](image)
understanding of the E45 connection assembly complexity is obtained (Table 9).

Among all, the highest score is associated with interface domain (4.12) and component domain (4.12). Analysing results for the interface domain, the attributes total length of harness and number of connections present a score of 5 (Table 10). Thus, the following alternative modifications might be implemented to reduce the interface domain score: (i) reduce the overall length of the connection by moving modules closer, (ii) merge modules together to minimize the connection (i.e. build a new module which encompasses the two modules or fit altogether the two modules outside the aircraft and bring them inside as a single module) and, (iii) make use of dedicated plate in order to install all connectors at the same time.

Moving to the component domain, attributes electrical weight and fragility present a score of 5. These scores are shared by all electrical interfaces. The reason behind this result lies on
| Name  | Type | Module IN | Module OUT | Interface domain |
|-------|------|-----------|------------|------------------|
|       |      |           |            | Total length of harness | Number of connections | Number of straight nodes |
| E45   | Elect| Module N  | Module S   | 5                | 5                 | 1                     |

| Name  | Type | Module IN | Module OUT | Ergonomic domain | Assembly domain |
|-------|------|-----------|------------|------------------|-----------------|
|        |      |           |            | Access          | Zone            | Working space size | Variety of tools | Process |
| E45.1  | Elect| Module N  | Module S   | 4               | 3                | 1                     | 3               | 4       |
| E45.2  | Elect| Module N  | Module S   | 4               | 5                | 1                     | 3               | 4       |
| E45.3  | Elect| Module N  | Module S   | 4               | 5                | 1                     | 3               | 4       |
| E45.4  | Elect| Module N  | Module S   | 1               | 3                | 4                     | 3               | 4       |

| Name  | Type | Module IN | Module OUT | Component domain | Electrical domain |
|-------|------|-----------|------------|------------------|-------------------|
|        |      |           |            | Electrical weight | Electrical place length | Fragility |
| E45.1.01 | Elect| Module N  | Module S   | 5                | 2                 | 5                     |
| E45.2.01 | Elect| Module N  | Module S   | 5                | 2                 | 5                     |
| E45.3.01 | Elect| Module N  | Module S   | 5                | 2                 | 5                     |
| E45.4.01 | Elect| Module N  | Module S   | 5                | 2                 | 5                     |
the fact that electrical connections, in the current architecture, are not divided into pieces like air interfaces; in fact same data are repeated for each interface sub-section (Table 11). Indeed, electrical harnesses are heavy and difficult to handle. Moreover, these harnesses are fragile mainly due to their lack of rigidity and they require special caution during installation operations. This aspect is represented by fragility score (i.e., 5). Thus, to improve the installation aspects of E45 and all other electrical interfaces, electrical harnesses shall be split into sub-harnesses and installed separately. However, the analysis performed does not consider the possibility to have shared interfaces, meaning interfaces sharing the same harness for a given length (or pre-bundled harnesses), which is typically the case for electrical harnesses.

The process can be repeated for the remaining domains (ergonomic domain and assembly domain) to identify potential improvements. The procedure is applicable to each interface for identifying criticalities during the installation phases. It is worth noting that the method does not consider the assembly sequence of the product (i.e., system dynamicity); thus some interfaces might be critical at the end of the installation process, but they might not be at an early stage of the process. Again, the hierarchical structure (framework) of the cBoM proposed within the CDFA methodology provides guidelines to the designers for analysing the product architecture and performing modification to improve the installation phase. In fact, by analysing the cBoM it is possible to highlight product issues related to the assembly phase.

### Table 11

| Name   | Type     | Module IN | Module OUT | Electrical weight | Electrical place length | Fragility |
|--------|----------|-----------|------------|-------------------|-------------------------|-----------|
|        |          |           |            | Original | Normalized | Original | Normalized | Original | Normalized |
| E5.1.01| Electrical| Module I | Module M   | 24.0     | 5          | 8.6      | 2         | High     | 5         |
| E5.2.01| Electrical| Module I | Module M   | 24.0     | 5          | 8.6      | 2         | High     | 5         |
| E5.3.01| Electrical| Module I | Module M   | 24.0     | 5          | 8.6      | 2         | High     | 5         |
| E8.1.01| Electrical| Module I | Module P   | 21.1     | 5          | 7.5      | 2         | High     | 5         |
| E8.2.01| Electrical| Module I | Module P   | 21.1     | 5          | 7.5      | 2         | High     | 5         |
| E8.3.01| Electrical| Module I | Module P   | 21.1     | 5          | 7.5      | 2         | High     | 5         |
| E17.1.01| Electrical| Module L | Module N  | 24.2     | 5          | 8.7      | 2         | High     | 5         |
| E17.2.01| Electrical| Module L | Module N  | 24.2     | 5          | 8.7      | 2         | High     | 5         |
| E17.3.01| Electrical| Module L | Module N  | 24.2     | 5          | 8.7      | 2         | High     | 5         |
| E19.1.01| Electrical| Module L | Module P  | 22.3     | 5          | 7.9      | 2         | High     | 5         |
| E19.2.01| Electrical| Module L | Module P  | 22.3     | 5          | 7.9      | 2         | High     | 5         |
| E19.3.01| Electrical| Module L | Module P  | 22.3     | 5          | 7.9      | 2         | High     | 5         |
| E45.1.01| Electrical| Module N | Module S  | 22.7     | 5          | 8.1      | 2         | High     | 5         |
| E45.2.01| Electrical| Module N | Module S  | 22.7     | 5          | 8.1      | 2         | High     | 5         |
| E45.3.01| Electrical| Module N | Module S  | 22.7     | 5          | 8.1      | 2         | High     | 5         |
| E45.4.01| Electrical| Module N | Module S  | 22.7     | 5          | 8.1      | 2         | High     | 5         |

### 6 Conclusion

Although DFA methods can reduce the assembly complexity of products, they are applicable only at late design phase, limiting their benefits. The CDFA method presented in this paper enables to assess assembly complexity of aircraft system architectures. The CDFA methodology and its general framework can be also adopted in the analysis of assembly performances of different products characterized by a high level of complexity (e.g., automotive, machinery and equipment industry, etc.). The method starts from the functional derivation of the product and, through the definition of levels, attributes and domains, creates the hierarchical structure of the cBoMs. Then, using specific mathematical operators, a single score for each interface is obtained. The highest score identifies the most complex interface to install. The method was applied to the nose-fuselage of a civil aircraft to assess the fit for assembly performance of the component represented with modules, electrical and air connections.

The novelty of the method is highlighted in the following points: (i) collecting and analysing data switching from tacit implicit knowledge to explicit knowledge (knowledge scoring matrices), (ii) create a mathematical model and a framework able to consider the global context of an aircraft assembly by using sub-problems that are limited in terms of complexity (problem discretization) and (iii) providing rationales to the designer regarding weakness of the product architecture in terms of assembly operations efficiency. Moreover, the method allows to perform comparisons of different system products architecture (e.g., nose-fuselage,
cabin, etc.) to understand which is better in terms of assemblyablity and to identify design actions aiming at improving system installation. The method does not directly provide redesign actions or solutions to designers, but it helps to spot product architecture issues.

However, the method presents some limitations as follows: (i) it does not offer a quick analysis of the modification that could be implemented to reduce the score: the solution space remains too wide to ease the down-selection of the most appropriate solution, (ii) it does not consider in the analysis the dynamic sequence of the operations and (iii) it does not consider interfaces of the same type which are shared.

Future works will focus on two aspects. On the one hand, an improvement of the CDFA methodology is required to overcome the existing constraints. For instance, since the current approach is not able to consider shared interfaces (i.e., one interface which is shared between more than one module), it will be necessary to strength the method to assess and identify this design solution. Moreover, an added value will be provided by the implementation of the methodology in commercial CAD systems to automatize the architecture scoring from the data acquisition phase to the scoring visualization. Nevertheless, it will be necessary to add a cost analysis; in fact, modularization might not be always the best solution since the interface cost might be too high to provide a real benefit (Engel and Reich 2015). On the other hand, research will focus on the developing of design assistance for exploring the solution space to improve the global product architecture scoring by implementing design guidelines. Regarding this aspect, an attempt has been done by the same authors to use the result of the CDFA method to drive the re-design of the cabin of an aircraft (Formentini et al. 2020, 2021). However, the proposed research is still on-going. Further research and testing are required to obtain valuable re-design suggestions.

Acknowledgements This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors.

References

Ahmed S (2005) Encouraging reuse of design knowledge: a method to index knowledge. Des Stud 26(6):565–592
Atasoy B, Safadi M, Bierlaire M, Leonardi C (2012) Impact analysis of a flexible air transportation system. Eur J Transp Infrastructure Res. https://doi.org/10.18757/ejtir.2013.13.2.2993
Barbosa GF, Carvalho J (2014) Guideline tool based on design for manufacturing and assembly (DFMA) methodology for application on design and manufacturing of aircrafts. J Braz Soc Mech Sci Eng 36(3):605–614
Bonvoisin J, Halstenberg F, Buchert T, Stark R (2016) A systematic literature review on modular product design. J Eng Des 27(7):488–514
Boothroyd G, Dewhurst P, Knight WA (2011) Product design for manufacture and assembly. CRC Press, Boca Raton
Borjesson FA (2010) Systematic qualitative comparison of five approaches to modularity. In: DS 60: proceedings of DESIGN 2010, the 11th international design conference, Dubrovnik, Croatia
Browning TR (2015) Design structure matrix extensions and innovations: a survey and new opportunities. IEEE Trans Eng Manag 63(1):27–52
Brusoni S, Prencipe A (2011) Patterns of modularization: the dynamics of product architecture in complex systems. Eur Manag Rev 8(2):67–80
Bullen GN (1999) Assembly automation and implementation issues. SAE Technical Paper 1999-01-2267
Butterfield J, Crosby S, Curran R et al. (2007) Optimization of aircraft fuselage assembly process using digital manufacturing. ASME J Comput Inf Sci Eng 7(3):269–275
Coma O, Masce C, Balazinski M (2004) Application of a fuzzy decision support system in a design for assembly methodology. Int J Comput Integr Manuf 17(1):83–94
Domeshek EA, Herndon MF, Bennett AW, Kolodner JL (1994) A case-based design aid for conceptual design of aircraft subsystems. In: Proceedings of the tenth conference on artificial intelligence for applications, IEEE, pp 63–69
El-Nounu A, Popov A, Ratchev S (2018) Redesign methodology for mechanical assembly. Res Eng Des 29(1):107–122
Engel A, Reich Y (2015) Advancing architecture options theory: six industrial case studies. Syst Eng 18(4):396–414
Engel A, Browning TR, Reich Y (2017) Designing products for adaptability: insights from four industrial cases. Decis Sci 48(5):875–917
Erens F, Verhulst K (1997) Architectures for product families. Comput Ind 33(2–3):165–178
Ericsson A, Erixon G (1999) Controlling design variants: modular product-platforms. Society of Manufacturing Engineers, Dearborn
Ethiraj SK, Levinthal D (2004) Modularity and innovation in complex systems. Manag Sci 50(2):159–173
Farid AM (2008) Facilitating ease of system reconfiguration through measures of manufacturing modularity. Proc Inst Mech Eng B 222(10):1275–1288
Favi C, Germani M (2012) A method to optimize assemblability of industrial product in early design phase: from product architecture to assembly sequence. Int J Interact Des Manuf 6(3):155–169
Favi C, Formentini G, Bouissiere F, Cuiller C, Dereux PE, Malchair C (2019) Design for assembly in the conceptual development of aircraft systems. In: International conference on design, simulation, manufacturing; the innovation exchange. Springer, Cham, pp 268–278
Formentini G, Favi C, Bouissiere F, Cuiller C, Dereux PE, Guillaume R, Malchair C (2020) Extrapolation of design guidelines during the conceptual design phase: a method to support product architecture design. In: Proceedings of the Design Society: DESIGN conference, vol 1. Cambridge University Press, pp 857–866
Formentini G, Cuiller C, Dereux PE, Bouissiere F, Favi C (2021) Impact assessment of design guidelines in the conceptual development of aircraft product architectures. Procedia CIRP 100:223–228
Frickenfielder S, Schulz AP (2005) Design for changeability (DIC): principles to enable changes in systems throughout their entire lifecycle. Syst Eng. https://doi.org/10.1002/sys.20039
Frigeri V, Talbot D (2005) Technological determinism and modularity: lessons from a comparison between aircraft and auto industries
