ORIGINAL PAPER

Model-based assembly process planning for flexible aircraft cabin architectures

Yassine Ghanjaoui[1] · Philip Satwan[1] · Jörn Biedermann[1] · Björn Nagel[1]

Abstract

The configuration and equipment of an aircraft cabin has a significant impact on the passenger's flight experience. To meet the needs of passengers and improve the flight experience, airlines repeatedly demand customised adaptations to the cabin design. The implementation of these changes requires a high degree of flexibility in production and can lead to difficulties in setting up a fixed final assembly line. In addition, unforeseen events and changes in the OEMs' supply chain require a rapid response in aircraft production to be able to deliver on time. Digitalisation in product development and production planning makes it possible to overcome these challenges and makes an important contribution to flexibility, time and cost efficiency. This paper presents a digital approach to modelling and flexible planning of assembly processes of the cabin. To this end, aircraft design data is automatically linked to the assembly system to react quickly to design changes in the assembly planning process. The integration is realised in a system architecture model depicted with the systems modelling language (SysML). The architecture model follows the formal process description defined in VDI3682. A planning algorithm then uses the production architecture parameters to optimise the assembly processes, e.g. in terms of time. The approach presented is demonstrated using the example of scheduling the pre-assembly processes of the "crown module". The latter includes all structural and functional components above the window panels, such as overhead bins, electrics and air conditioning. The results show how changes in aircraft and cabin design or in the production plant and resources can be flexibly evaluated. This allows conceptual changes to be evaluated and traded before the cabin design is frozen and transferred to real production. These advantages contribute to the time and cost optimisation of aircraft production. This work thus makes a valuable contribution to the further development of digital solutions in aircraft production.

Abbreviations

API	Application Programming Interface
CAD	Computer Aided design
CDfA	Conceptual Design for Assembly
CPACS	Common Parametric Aircraft Configuration Schema
CPPS	Cyber-Physical Production System
DLR	Deutsches Zentrum für Luft- und Raumfahrt
FUGA	Fuselage Geometry Assembler
INCOSE	International Council of Systems Engineering
JSON	JavaScript Object Notation
KBE	Knowledge-Based Engineering
MBSE	Model-Based Systems Engineering
OHSC	Overhead Storage Compartment
PPR	Product Process Resource
SysML	Systems Modelling Language
TLAR	Top Level Aircraft Requirements
VDI	Verein Deutscher Ingenieure
XML	Extensible Markup Language

✉ Yassine Ghanjaoui
  yassine.ghanjaoui@dlr.de

  Philip Satwan
  philip.satwan@dlr.de

  Jörn Biedermann
  joern.biedermann@dlr.de

  Björn Nagel
  bjoern.nagel@dlr.de

1   Institute of System Architectures in Aeronautics,
    German Aerospace Center (DLR), Hein-Saß-Weg 22,
    Hamburg 21129, Germany

1 Introduction

Aircraft development is a complex and dynamic environment, from the design phase to industrialisation and production. For each aircraft, engineers in different domains need to manage the high variation of requirements, resources, materials and tools [1]. It is estimated that over 30% of the total cost of an aircraft is incurred during production, with 80% of these costs being determined during the concept design phase, where engineers have more freedom in their design decisions [2]. Today, aircraft design and production planning are still seen as two separate engineering competencies, characterised by highly interdependent tasks with a clear division of labour and application of specific models, methods and tools [3]. This separation makes it difficult to rapidly evaluate the integration of new technologies such as hydrogen systems or fuel cells in early development phases. Especially in the aircraft cabin, a high degree of flexibility, modularity and reconfigurability are necessary due to the individual requirements of the airlines. To meet these requirements, the manufacturing industry must be able to adapt its processes to the selected products and bring them to market at minimum cost and in the shortest possible time, taking into account quality and flexibility [4].

Currently, the assembly processes in the cabin are firmly defined and predominantly manual. This means that workers are unable to react to sudden changes [4, 5]. Digitalisation creates the opportunity to integrate the dynamic factors in a digital environment to link aircraft production with design and optimise it across disciplines. The so called digital thread that links design and production will become the core element of the factory of the future [6]. For these reasons, researchers at the German Aerospace Centre (DLR) are developing methods to establish digital continuity throughout the entire product life cycle and accelerate the development of sustainable aviation. These methods aim to link conceptual aircraft design with production planning and manufacturing systems to plan, evaluate and optimise the assembly and production of new or customised aircraft and cabin configurations. In this way, they provide feedback from production to design and enable cost-efficient design optimisation at an early stage. In addition, the optimal utilisation of available resources such as robots and workers for the required assembly or installation processes can be planned, simulated and tested in a flexible way through the use of production models.

In this context, this work presents an approach that utilises various digital methods and technologies to support the semantic linking of cabin design with production planning. The approach combines knowledge-based tools for creating a cabin design with Model-based Systems Engineering (MBSE) methods. As a result, information is semantically linked and the assembly processes are formally described. This work provides a solid foundation for the semantic, model-based integration of product and production development, contributing to DLR's ongoing efforts to advance digital and sustainable aviation.

2 State of the art and methodological requirements

As there is no series production in cabin assembly, it is difficult to optimise assembly in advance through physical prototypes or pre-söeries production [7]. In addition, customer-specific requirements for individual and customisable cabin designs require a high degree of flexibility in cabin production and assembly. A common definition of the cabin architecture and production network should be used to anticipate customer changes and ensure their traceability throughout the product development process [8]. In the aviation industry, the design and production planning of a cabin product are currently carried out separately: the design engineers first create a design before the production engineers create the production plan and provide feedback on possible changes to facilitate the assembly process. The design engineers then redesign the product to fulfil these requirements in an iterative process. This cycle is repeated until a competitive design is achieved [9]. This makes it difficult to provide the required flexibility between cabin design and production and alternative approaches to flexible design and planning are required.

In [10] a method was developed that uses ontologies of product and production system for automated process planning of cabin assembly. The ontology was implemented in a scheduling algorithm that agilely and automatically reschedules the cabin assembly processes and reduces additional expert effort. The system architecture is integrated into the ontology with the algorithm, which requires ontology expertise from the system architect. In addition, the cabin design is not automatically linked to the ontology. Similarly, in [11] a common cabin metamodel with the underlying ontologies was developed to describe the product and assembly process planning. Relationships between model elements of different domains and life phases is the main advantage of the metamodel as it enables traceability and error checking. In addition, the metamodel provides the ability to establish a common data source, which is an important factor when data changes rapidly, as in the early stages of cabin co-design. However, the instantiation of the metamodel is primarily used for the description of processes with linked products and resources and not for automated assembly planning and optimised allocation of processes to resources. Another method in [12] supports the integration of new cabin

concepts into the cabin. The use of detailed Computer Aided Design (CAD) data prevents the computation of conceptual planning using the parametric relationship between design and assembly. Finally, [13] presents a Conceptual Design for Assembly (CDfA) methodology for the cabin architecture concept of a commercial aircraft. Although critical points of the design during assembly are identified, these results are not integrated into the planning of optimised assembly under consideration of the design adjustments.

The recognised deficiencies in the methods presented are to be expanded by an approach to assembly planning for flexible cabin architectures. The requirements for this approach are defined as follows:

- **R1:** Conceptual cabin design and assembly should be linked parametrically
- **R2:** The link should lead to automated assembly planning for flexible cabin architectures.
- **R3:** Individual cabin product designs should be automatically evaluated from an assembly perspective.
- **R4:** Efficient allocation of assembly processes to available resources should be based on process requirements and resource capabilities.

Therefore, a knowledge- and model-based methodology is developed in this work to fulfil the requirements and is presented in the following section.

3 Methodology for model-based assembly planning of aircraft cabins

To achieve the goal of flexible and responsive production planning, a model-based methodology that semantically links cabin design with assembly planning was developed. Digital technologies, semantic guidelines and knowledge-based tools are used for this purpose. The most important methodological steps and the technical tools used for this are presented in Fig. 1.

The methodology distinguishes between knowledge-based cabin design, product-process-resource system architecture and knowledge-based assembly process planning. The required flexibility in assembly planning is achieved through data consistency and semantic linking between these artefacts. The artefacts mentioned are explained in the following sections.

3.1 Knowledge-based cabin design

The first step is to generate a cabin design from a set of requirements and provide the associated information for assembly planning. The Fuselage Geometry Assembler (FUGA) tool developed by DLR is used to generate a design parametrically. FUGA was developed using a parametric, knowledge-based methodology to track the effects of the design in all phases of the design and at the same time implement changes quickly and rule-based. It enables a consistent and parametric design of the fuselage and cabin. In this way, FUGA is integrated into the overall workflow and combines preliminary design and process planning.

FUGA uses the Common Parametric Aircraft Configuration Schema (CPACS) as the central model for the input and output data structure, which contains the product data [14–16]. The CPACS dataset contains a preliminary aircraft design based on the Top Level Aircraft Requirements (TLAR). This creates the initial boundaries for the subsequent design, which is performed with FUGA. The CPACS dataset does not contain any information about the cabin interior. To enable a customised cabin layout, some additional design parameters can be entered in a JSON dataset (e.g. the passenger arrangement or the specific placement of cabin monuments) [16]. This set of inputs is then used as the basis for the subsequent design step. There, FUGA follows a Knowledge-based Engineering (KBE) approach to generate a finished design from a consistent parametric basis, into which additional data (e.g. the cabin interior) is incorporated. The resulting cabin design is finally structured

Fig. 1 Methodology for model-based assembly planning of the aircraft cabin
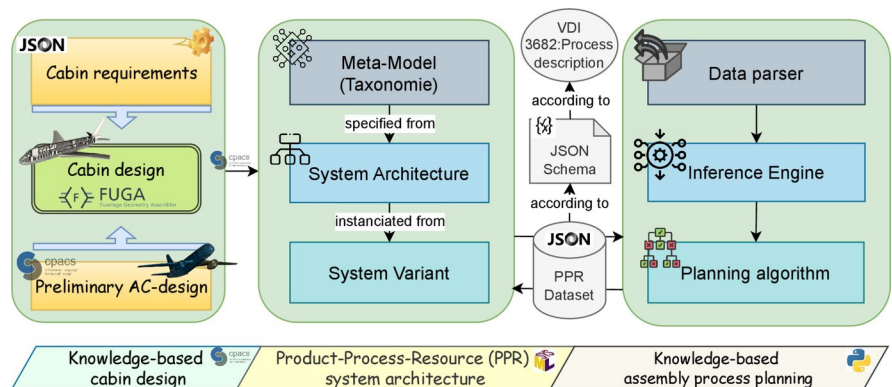
in CPACS format. A more detailed description of the methodology behind FUGA is given in [16].

3.2 System architecture for products, processes and resources

The methodology follows the MBSE approach to conceptually map the system architecture and apply it as a basis for production planning. Given the complexity of the production planning task, MBSE can provide a suitable framework for decomposing and analysing the interactions of abstract artefacts, enabling a systemic and holistic approach to meeting the numerous requirements. MBSE provides numerous tools, methods and guidelines that support the specification, analysis and development of complex systems whilst ensuring the consistency and traceability of modelling artefacts in the MBSE context [17, 18]. The System Modelling Language (SysML) [19] supports the modelling activities and is used in this work. The first step is to define a metamodel that provides a taxonomy for modelling. When defining the taxonomy, it is important to refer to existing standards and guidelines that support the representation of essential concepts and relationships in a particular domain. Therefore, the created metamodel references the formalised process description for Product Process Resource (PPR) as specified in the VDI guideline 3682 [20]. The latter provides visual representations to enable a comprehensive model of the functional view of the Cyber-Physical Production System (CPPS) in the initial planning phase [21]. This knowledge representation can be used to describe how products, processes and resources interact to achieve the defined production functions and objectives. An overview of the metamodel depicted in SysML is shown in Fig. 2.

These concepts are modelled using block definition diagrams (bdd) and SysML relationships. The decomposition relationship can be used to represent the concept of a process consisting of an undefined number of *parts* of type *ProcessOperator*. This is intended to represent the concept of assembly processes that have inputs and outputs of types *Product*, *Energy* and *Information*. Each of these elements is represented by a block with specific attributes. To model the relationship between these blocks, reference properties are used that are typed by the related elements, e.g. to show that a resource is assigned to a process operator. The system context of a superordinate process can also be specified with the reference of type *SystemBorder*. Attributes that are relevant for the process analysis are specified as block value properties. Further specifications, such as the distinction between different types of resources or products, are modelled using generalisation relationships.

The described metamodel with the PPR concepts can be used in the definition of the cabin system architecture and linked to cabin-specific architecture blocks via generalisation relationships. An important part of the architecture is the definition of restrictive relationships between the assembly and the design. To do this, abstract requirements are first defined and linked to logical conditions in the form of *constraints* elements, which in turn are linked to the PPR components. The data of a specific configuration are read from the CPACS data set and used to instantiate the cabin architecture in the SysML model. The defined constraints help to restrict the instantiation and automatically apply it to the required framework conditions. As a result, linked instance data on the cabin products, the available assembly resources and the corresponding processes are available in the architecture model.

Fig. 2 Overview of the taxonomy definition as SysML metamodel related to formal process description guideline VDI3682
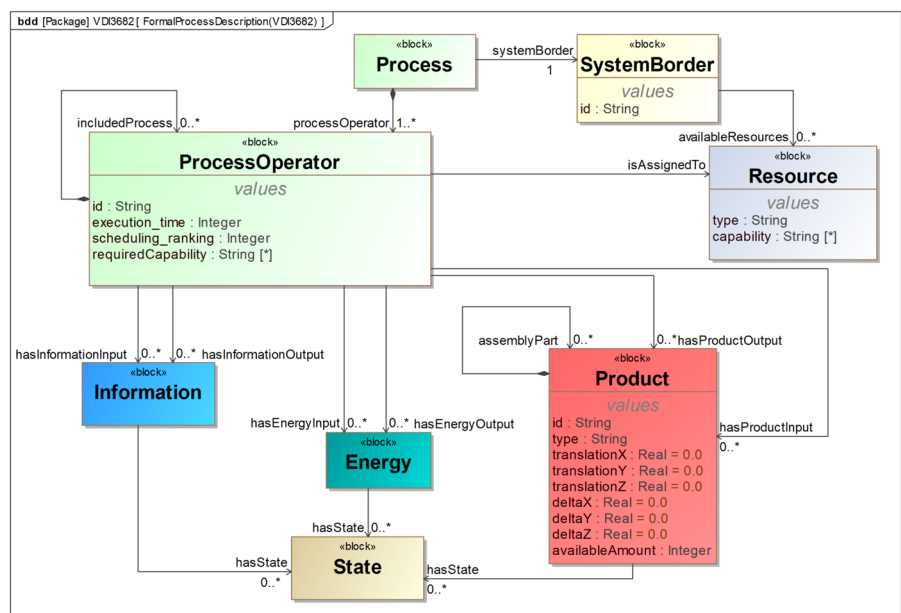
In the current implementation of data linking between the PPR architecture and the knowledge-based assembly process planning, no extensive data management system has been developed yet. Currently, architecture results are saved on a local server in a JSON file, which is accessed by the knowledge-based software for further processing. The JSON file is created automatically in the SysML Tool (Cameo Systems Modeler)[1] by querying information from the model elements using their corresponding types as defined in the VDI 3682 taxonomy description (see Fig. 2). Once the planning activities are completed, a new file following the same schema is created and stored in the same repository. This new file includes additional information but retains the previous data and can be accessed for further architecting activities. For sophisticated data storage and exchange within company IT infrastructures, a proper data management system is needed to enable effective data versioning, access, and documentation.

3.3 Knowledge-based assembly process planning

The exchange of PPR instance data is based on a consistent, standardised, validatable and traceable data structure described using a JavaScript Object Notation (JSON) schema. JSON is a compact data format in an easy-to-read text form for exchanging data between applications. A JSON schema is a standard that provides a format for what JSON data are required for a particular application and how to interact with it [23]. The schema corresponds to the formal process description of VDI 3682 and enables assembly process planning algorithm to read PPR information from the JSON data set. These are the processes to be executed, available resources, resource capabilities, product availability, product position and expected process duration. Additional information required for planning the assembly processes can be automatically interpreted from this data using an inference engine. For example, processes are associated with resources based on capabilities or the process dependencies and product structure can be interpreted.

The information read or inferred from the architecture represents the knowledge base used by the planning algorithm. The task is to flexibly sequence the assembly processes for a specific configuration of cabin components and available resources and to optimally allocate them to the resources. Several approaches for planning machine orders can be found in the literature. In [24], for example, an approach for planning orders for a single machine with

maintenance activities is presented in which a genetic algorithm solves the planning activities. Disadvantages of this algorithm are its computational intensity and that there is no guarantee for a global optimum, which complicates the planning and validation of complex assembly tasks. Authors in [25] present an integrated approach to work scheduling and assembly sequence planning for discrete manufacturing, which can be used to optimise the machining sequence of parts and the assembly sequence simultaneously. These are suitable for the planning task but are characterised by complexity and limited applicability for different assembly scenarios.

Due to its simple structure, good traceability and interpretability, a decision tree algorithm was used in this work. This algorithm uses a tree-like flowchart that starts with a *root node* and continues with *leaves*. The branches are generated as the result of a feature-based split. The conditions for splitting the node are if-else statements, and a split action can be defined if the statement is true or false. In real applications, the data sets have a large number of features, which leads to a large number of splits. These larger trees require a lot of computing power to compute and have high accuracy on the training data and poor accuracy on the test data. This is called overfitting, and to avoid this, a pruning procedure is needed, in which nodes with low importance are removed during the growth of the tree [26].

The approach presented in this work utilises the decision tree algorithm developed and validated by the authors in [10]. No systematic changes have been made to the algorithm itself; it has only been adjusted to fit the parameters of the presented use cases. This algorithm is grounded in methods found in the literature [27]. In this work, the plausibility of the results generated by the algorithm is evaluated through an analysis of the implemented use cases and the resulting assembly process sequence. The primary focus of this study is not on developing or assessing the algorithm itself. Instead, it is used as a tool for process planning and scheduling, with the emphasis on integrating it into the overall methodology. This includes linking it to knowledge-based modules, system architecture, and cabin design. Therefore, this research does not provide a comprehensive validation and assessment of the decision tree algorithm.

The decision tree provides the assembly process sequence and corresponding assignment to the available resources as output. This information is specified in the JSON data set and made available for further usage, such as process simulation or resource control.

---

[1] Cameo Systems Modeler is a collaborative Model-Based Systems Engineering (MBSE) environment, which provides tools to define, track, and visualise all aspects of systems in a standard-compliant SysML models and diagrams. [22]

4 Application results in the context of cabin assembly

4.1 Application case: Pre-assembly of the crown module

In this Section, a use case is presented to assess the methodology presented above and its practical applications. The application used in this work is based on the pre-assembly process of the so-called "crown module" (see Fig. 3a), which is currently a research focus in the installation of aircraft cabins to decouple the cabin from the airframe. The crown module comprises the ceiling area of the aircraft, the Overhead Storage Compartment (OHSC) and a variety of customised components from cabin systems including electrical supply, cabin communication, air conditioning and oxygen supply. Due to the complexity and variety of these components, the individual positioning and assembly of each component is a time-consuming process. Moreover, there are numerous interfaces between them and the primary structures, which further increases the complexity. By developing a pre-assembled module that integrates the cabin components and can be installed in the aircraft in a single assembly step, it is possible to decouple the cabin from the airframe. The purpose of this decoupling is to segregate the standardised 'make-to-stock' airframe production from the customised 'assembly-to-order' cabin assembly process. The planning of the pre-assembly is realised for the pre-assembly station within the DLR laboratory (see Fig. 3b). Available resources that are included in the use case are an industrial robot UR10e from Universal Robots [28], an APAS cobot from Bosch [29], workers and a pneumatic crown module jig.

As part of this application, two use cases are defined that serve as a proof of concept (PoC) for the requirements of the methodology. In the first use case, the flexibility of assembly planning for configurable cabin architectures and for the modular use of resources is verified. The assembly for the cabin architecture and resource availability is evaluated based on the assembly time. In the second use case, the assembly planning methodology for flexible architectures of the products to be assembled is verified. The implementation and corresponding results for the two use cases are presented in the following sections.

4.2 Configurability of the cabin and resource architecture

As already mentioned, the early investigation of novel, sustainable technologies in the aircraft is of great interest. Therefore, two different aircraft configurations with hydrogen tanks were investigated in this first case. During the conceptual phase, it is important to not only analyse the design, but also evaluate the impact of the tank integration on the cabin assembly. As already explained in Sect. 3, the cabin design parameters for a selected preliminary design and specific cabin requirements are generated by FUGA. The two cabin designs are shown in Fig. 4.

The first configuration with larger tanks leads to a shortening of the cabin area compared to the second configuration. As a result, the positions of the door areas and the number of frames vary, which affects the architecture of the crown module. The design parameters for the respective cabin configuration in CPACS format are linked to the system architecture using *value properties* in the parametric diagram (see Fig. 5).

The CPACS nodes with the information regarding the door positions and distance between frames are parsed for this purpose. In the SysML model, *constraints* are defined to fulfil the assembly requirements, which calculates assembly parameters such as the number and positions of the crown module sequences to be pre-assembled from the design information. Two architecture instances are automatically generated for the respective configuration. The availability of resources can also be set for each instance. The tables in Fig. 6 and 7 show the two selected resource configurations.

In the next step, the PPR architecture information available in the SysML model is exported as a JSON data set formatted according to the already defined schema (see Sect. 3) and used by the algorithm to plan the crown module

Fig. 3 Set-up for the application case

(a) Crown module [30]

(b) Pre-assembly station of the DLR

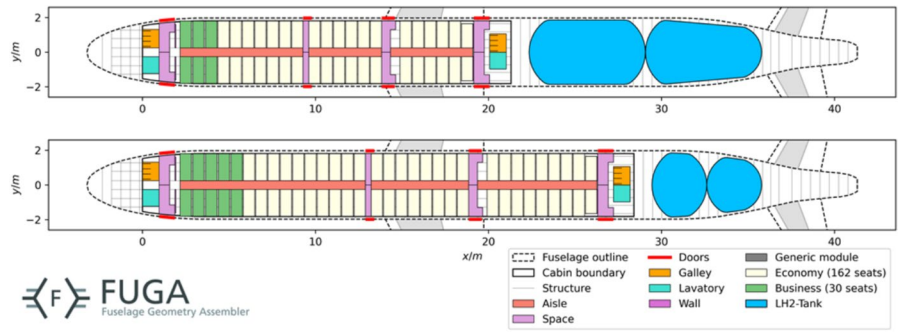**Fig. 4** Results of the cabin design for two hydrogen configurations generated by FUGA

Fig. 5 Parametric diagram for linking the system architecture with the cabin design parameters in CPACS
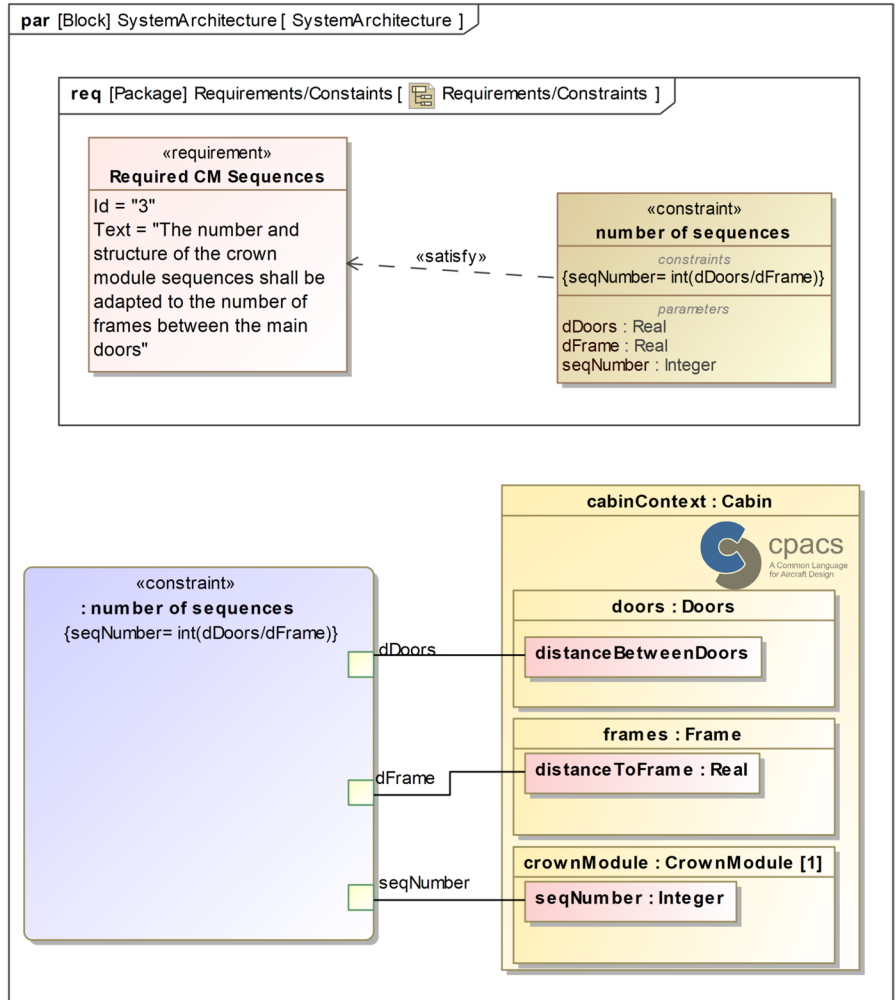

Fig. 6 Resource configuration with a worker and a UR10 robot

#	Name	resource : Resource	type : String	capability : String
1	resourceList	Human_1 : Human UR10_1 : Human		
2	Human_1		Human	pick_and_place
3	UR10_1		UR_10	fastening_with_bolt_and_nut

Fig. 7 Resource configuration with a worker, a UR10 robot and an APAS robot

#	Name	P resource : Resource	V type : String	V capability : String
1	resourceList	Human_1 : Human UR10_1 : Human		
2	Human_1		Human	pick_and_place
3	UR10_1		UR_10	fastening_with_bolt_and_nut
4	APAS_1		APAS	fixing_in_position fastening_with_bolt_and_nut

Fig. 8 Results of the assembly process planning for the two hydrogen configurations using two resource variants

pre-assembly. The algorithm provides the process sequence, pre-assembly time and the allocation of processes to the available resources. These results for the centre cabin section are shown in Fig. 8. The assembly time is represented in time units and the assembly processes carried out by the respective resource are shown.

The computing time required for the decision tree algorithm to generate assembly sequences has increased significantly (by 20%) after adding one available resource. The allocation of assembly processes to the APAS robot made it possible to parallelise these processes and reduce the assembly time for both configurations. Therefore 5 time units have been reduced for the first cabin configuration and 7 time units for the second one. Moreover, the second configuration includes two more sequences due to the larger areas between the cabin doors. This leads to an increase in assembly time because of the required additional processes. Interestingly, the first configuration

without the APAS robot requires a comparable assembly time (20 time units) to the second configuration with the APAS available (19 time units). Thus, the negative impact of the cabin architecture on the assembly time can be offset by modifying the resource configuration.

The results also show the dependencies between the assembly processes and the restrictions imposed by the resource capabilities. In all configurations, the fastening processes (*fastening*) of the respective crown module sequence only start after they have been picked up and placed (*pick and place*). The *idle* processes result due to the fact that the respective resource does not have the necessary capability to perform a certain process. For example, in the first configuration without the APAS robot, the worker cannot assist with the fastening processes in parallel with the UR10 robot because the required capability has not been assigned to it (see Fig. 6).

4.3 Configurability of the product architecture

In the second application case, the architecture influence of the product to be assembled, in this case the crown module, on the assembly process is analysed. With the proposed approach, changes in the product architecture can be introduced directly in the SysML model and lead to new constraints and dependencies for assembly process planning.

In addition, modularisation concepts of the product can be analysed and evaluated with regard to assembly properties.

To model the crown module architecture, its hierarchical structure is first mapped using a *block definition diagram (bdd)*. Figs. 9 and 10 depict two different crown module structures in their respective bdd.

The modularisation concept can be implemented using composition relationships. Each module is modelled as a

Fig. 9 block definition diagram with first crown module structure
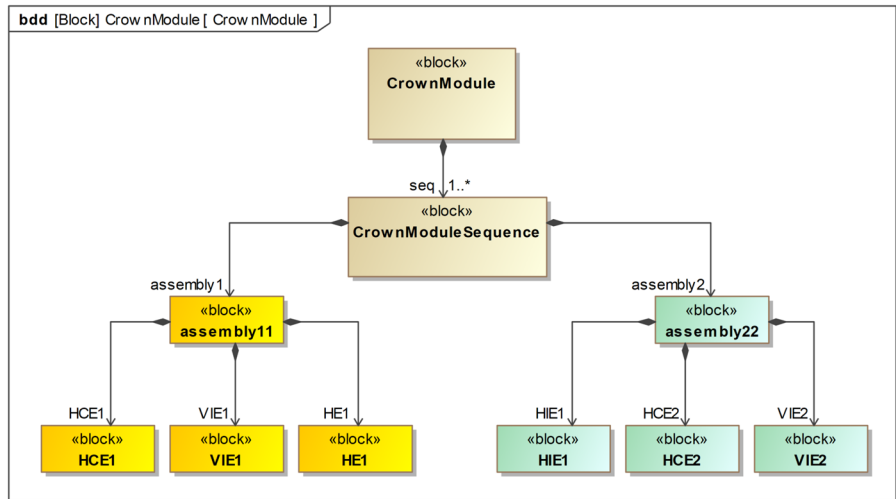
Fig. 10 block definition diagram with second crown module structure
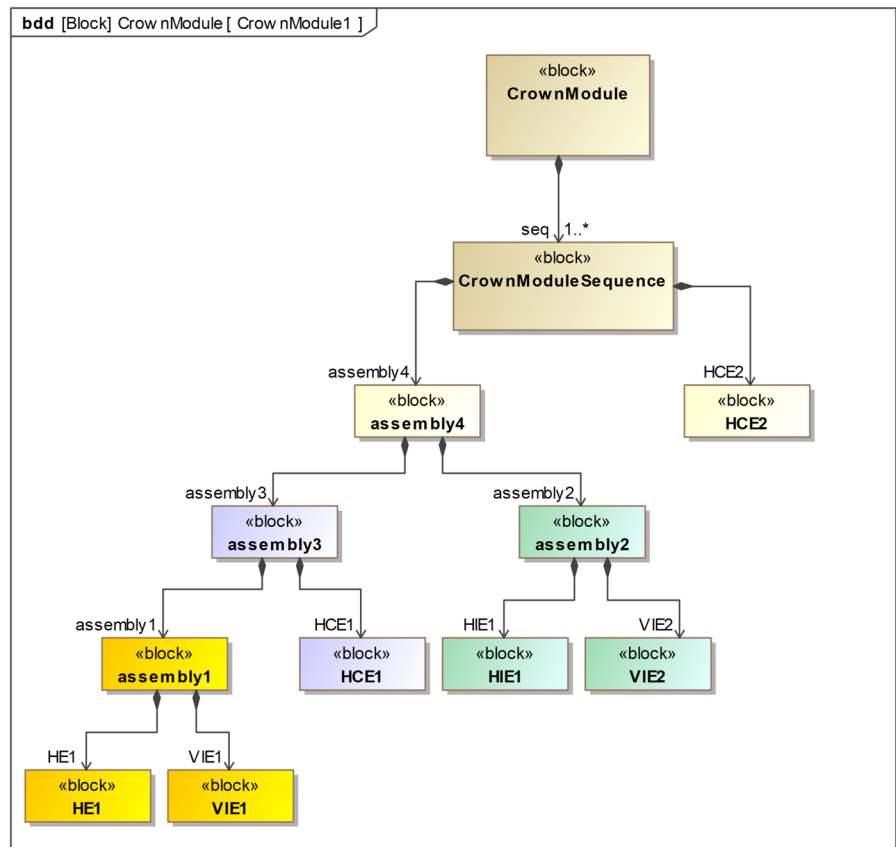
sub-assembly block (*assembly*) that groups the module components. These can only be assembled with other external components after module assembly. This results in assembly restrictions from the product architecture. In addition to the hierarchical structure, the interfaces between components and modules are defined in the *internal block diagram (ibd)*. By connecting the interfaces with *connectors*, assembly relations between the module components can be specified.

The planning algorithm is executed for the two crown module architectures presented in Figs. 9 and 10. In this case, two human workers are available with both the capability to pick and place and to attach. The results are shown in Fig. 11.

The SysML elements for describing the two architecture variants, such as composition relationships or interface specifications, are used to generate the formal process description of the assembly processes. They define input and output products for the respective processes, which are interpreted by the planning algorithm as constraints in the next step. This constraint restricts which components must be assembled before others and thus leads to different assembly scenarios. The results show that, despite the same crown module components to be assembled and using the same resources, different assembly times result. The first crown module variant has a flat hierarchy with two compact modules. This enables parallel assembly of the two modules and subsequent assembly of the two pre-assembled modules. In contrast, the steeper hierarchy leads to more complex dependencies and more assembly interfaces. The higher number of modules to be pre-assembled increases the assembly time, as the assembly can only take place sequentially due to the restrictions and the second worker is therefore not assigned to any process for 12 time units.

These results show the impact of the product architecture on the assembly time. In this case, only the modularisation is evaluated as an architecting mechanism. Other architecture variation aspects such as different component properties or interfacing technologies can have a significant influence on assembly and need to be evaluated in future research.

5 Benefits and limitations

The application of the methodology based on the crown module use case allows for the requirements defined in Sect. 2 to be verified. The integration of the design parameters from the knowledge-based design into the PPR architecture makes it possible to visualise the interactions between the cabin configuration, the product architecture and the resulting assembly planning. The use of constraints to link the design parameters in CPACS with the assembly processes in a consistent PPR architecture enables to take into account the production requirements for integrated cabin configurations. This fulfils requirement (**R1**) and allows different cabin configurations and product architectures to be evaluated from an assembly perspective as early as the conceptual phase.

The automated export of the instantiated architecture in a standardised, formal description format provides the semantic relations in the PPR architecture for algorithmic assembly planning. The selected JSON format has the advantage of simple serialisation and reading of the information from the consistent and validatable data set. This information includes not only PPR parameters and attributes such as process duration and product positions, but also dependencies and restrictions for assembly planning. These were shown, for example, by the individual product modularisation. The semantic relations enable the automation of assembly planning for the individual products and thus the requirements (**R2** and **R3**) could be verified.

Furthermore, the decision tree algorithm enables defined boundary conditions to be taken into account when

Fig. 11 Results of the assembly process planning for the two crown module configurations with the availability of two workers
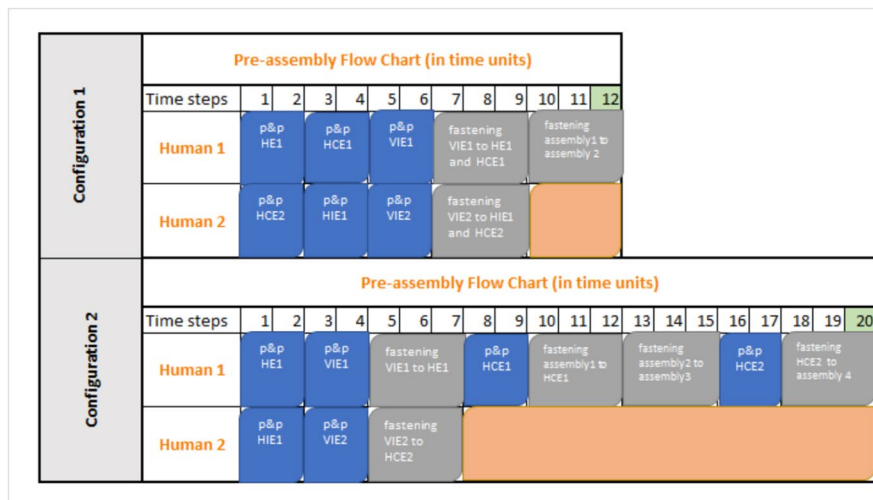
sequencing the assembly processes. The boundary conditions are interpreted in the algorithm by the inference engine and thus additional information can be formalised that is relevant for assembly process planning. A query is performed in the nodes according to the capabilities and availability of the resources and thus leads to an efficient allocation and process assignment for the resources. This means that both the process sequencing and the resource allocation can be planned to suit the cabin architecture and thus fulfil the last requirement (**R4**).

However, the application also shows limitations of the presented methodology. When modelling the PPR architecture, simplifications and assumptions were made when defining certain parameters. The required processes were specified directly and were not methodically derived from the product architecture. Similarly, important parameters such as process duration and resource capabilities are defined by assumptions. These should be defined by real hardware data or validatable simulation models, taking model uncertainty into account. In addition, although the information contained in the PPR architecture can be formalised in JSON format, data modelling with the format can be limited for more complex architecture models and for the concept extension. The schema definition must be constantly adapted and more complex logical relations cannot be formalised with the format. In addition, the execution of the planning algorithm resulted in a significant increase in calculation time with an increase in the complexity of the PPR architecture. This can be a clear limitation for the planning of assembly for more complex cabin products and systems as well as for an extension of the resource model. Lastly, the planning was only carried out for one indicator, in this case the assembly time. As a rule, the planning and optimisation of the design should be implemented multifactorially for multidisciplinary indicators (e.g. material flow or energy consumption). The recognised limitations and disadvantages are to be considered in further research by extending and adapting the methodology presented.

6 Summary and outlook

The demand from airlines for customised and adaptable cabin designs requires a high degree of flexibility in assembly planning. To ensure that the corresponding processes can be flexibly adapted to changes in cabin architecture, cabin design and assembly must be closely integrated. As part of this work, a model-based methodology was developed that semantically links conceptual cabin design parameters with assembly planning artefacts.

To this end, design parameters were integrated to instantiate a SysML-based PPR system architecture. A JSON-schema was developed to structure and formalise the system architecture in accordance with the formal process description given in VDI3682. A decision tree algorithm was also developed for efficient cabin assembly planning. This algorithm uses existing PPR information as well as knowledge-based interpretable relations and constraints. The methodology was applied and verified using the pre-assembly of the crown module. Two use cases for the configurability of the architecture of the cabin, the resources and the product to be assembled were implemented.

The results showed that the methodology enables automated planning and evaluation of different cabin configurations from an assembly perspective. The concept of modularisation of the product architecture was used to demonstrate the flexibility and assessability of assembly planning for different product variants. It was also verified that knowledge-based assembly planning enables efficient allocation of processes to available resources based on process requirements and resource capabilities. Limitations were found in the formalisation of more complex and extensible PPR architectures with the developed JSON-schema. High computation times of the algorithm were also recognised for more complex PPR datasets. In future research, these limitations are to be eliminated through extendable formalisation concepts. An optimisation approach is also to be integrated into the methodology to take more complex PPR architectures into account and optimise them according to several indicators.

Declarations

References

1. Vaidya, S., Ambad, P., Bhosle, S.: Industry 4.0 - a glimpse. Procedia Manuf. **20**, 233–238 (2018). https://doi.org/10.1016/j.promfg.2018.02.034
2. Raju, J.: A conceptual design and cost optimization methodology. In: 44th AIAA/ASME/ASCE/AHS/ASC structures, structural dynamics, and materials conference. American Institute of Aeronautics and Astronautics, Reston, Virigina (2003). https://doi.org/10.2514/6.2003-1505
3. Sinnwell, C., Krenkel, N., Aurich, J.C.: Conceptual manufacturing system design based on early product information. CIRP Annals **68**(1), 121–124 (2019). https://doi.org/10.1016/j.cirp.2019.04.031
4. Ottogalli, K., Rosquete, D., Rojo, J., Amundarain, A., Rodríguez, J.M., Borro, D.: Framework for the simulation of an aircraft final assembly line. MATEC Web of Conferences **233**, 00010 (2018). https://doi.org/10.1051/matecconf/201823300010
5. Ottogalli, K., Rosquete, D., Rojo, J., Amundarain, A., María Rodríguez, J., Borro, D.: Virtual reality simulation of human-robot coexistence for an aircraft final assembly line: process evaluation and ergonomics assessment. Int. J. Comput. Integr. Manuf. **34**(9), 975–995 (2021). https://doi.org/10.1080/0951192X.2021.1946855
6. Henke, R.: An der Schwelle zu einem neuen Luftfahrt-Zeitalter. DLR Magazin, Deutsches Zentrum für Luft- und Raumfahrt e.V. **457**, 4 (2021)
7. Heinig, M.: Nutzung von virtuellen Technologien für die Montageplanung von unikaten. PhD thesis, Hamburg, Technical University Hamburg-Harburg (2015)
8. Buergin, J., Belkadi, F., Hupays, C., Gupta, R.K., Bitte, F., Lanza, G., Bernard, A.: A modular-based approach for just-in-time specification of customer orders in the aircraft manufacturing industry. CIRP J. Manuf. Sci. Technol. **21**, 61–74 (2018). https://doi.org/10.1016/j.cirpj.2018.01.003
9. Page Risueno, J., Nagel, B.: Development of a knowledge-based engineering framework for modeling aircraft production. In: AIAA Aviation 2019 Forum. American Institute of Aeronautics and Astronautics, Reston, Virginia (06172019). https://doi.org/10.2514/6.2019-2889
10. Markusheska, N., Srinivasan, V., Walther, J.-N., Gindorf, A., Biedermann, J., Meller, F., Nagel, B.: Implementing a system architecture model for automated aircraft cabin assembly processes. CEAS Aeronaut. J. **13**(3), 689–703 (2022). https://doi.org/10.1007/s13272-022-00582-6
11. Berschik, M.C., Blecken, M., Kumawat, H., Rath, J.-E., Krause, D., God, R., Schüppstuhl, T.: A holistic aircraft cabin metamodel as an approach towards an interconnected digitised cabin lifecycle. In: 33rd Congress of the International Council of the Aeronautical Sciences, ICAS (2022). https://doi.org/10.15480/882.4757.
12. Halfmann, N., Krause, D., Umlauft, S.: Assembly concepts for aircraft cabin installation. ASME 2010 10th Biennial Conference on Engineering Systems Design and Analysis, **4**, 733–739 (2010). https://doi.org/10.1115/ESDA2010-24816
13. Formentini, G., Favi, C., Bouissiere, F., Cuiller, C., Dereux, P.-E., Jurbert, C.: A method to assess design for assembly efficiency of aircraft cabin concepts. Design Tools and Methods in Industrial Engineering II, 287–297 (2022). https://doi.org/10.1007/978-3-030-91234-529
14. Marko Alder, Erwin Moerland, Jonas Jepsen, Björn Nagel: Recent advances in establishing a common language for aircraft design with cpacs. In: Aerospace Europe Conference (2020)
15. Walther, J.-N., Hesse, C., Alder, M., Biedermann, J.Y.-C., Nagel, B.: Expansion of the cabin description within the cpacs air vehicle data schema to support detailed analyses. CEAS Aeronaut. J. **13**(4), 1119–1132 (2022). https://doi.org/10.1007/s13272-022-00610-5
16. Walther, J.-N., Hesse, C., Biedermann, J., Nagel, B.: Extensible aircraft fuselage model generation for a multidisciplinary, multifidelity context. 33rd Congress of the International Council of the Aeronautical Sciences (ICAS) (2022)
17. Walden, D.D., Roedler, G.J., Forsberg, K.: Incose systems engineering handbook version 4: Updating the reference for practitioners. INCOSE Int. Symp. **25**(1), 678–686 (2015). https://doi.org/10.1002/j.2334-5837.2015.00089.x
18. Kapurch, S.J.: NASA Systems Engineering Handbook, 1st edn. NASA, (2010)
19. ObjectManagementGroup: SysML. http://www.omgsysml.org/ [visited 28.03.2024] (2023)
20. Lars Christiansen, Tobias Jäger, Frank Schumacher, Alexander Fay: Modellierungsvorschlag zur grafischen Beschreibung alternativer und paralleler Prozessabläufe auf Basis eines Vergleichs bestehender Beschreibungsmittel (2012)
21. Meixner, K., Decker, J., Marcher, H., Luder, A., Biffl, S.: Towards a domain-specific language for product-process-resource constraints. 2020 25th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA) (2020). https://doi.org/10.1109/etfa46521.2020.9212063
22. DassaultSystemes: Cameo systems modeler The Software tool for systems engineering and modeling. https://www.3ds.com/products/catia/no-magic/cameo-systems-modeler [visited 01.08.2024] (2024)
23. Organization, T.J.S.: Why JSON Schema. https://json-schema.org/ [visited 28.03.2024] (2020)
24. Esmaeili, M.: Optimization costs of the single-machine scheduling problem with maintenance activities by using genetic algorithm. Manag. Sci. Lett. **2**(2), 673–680 (2012). https://doi.org/10.5267/j.msl.2012.10.014
25. Wang, Z.-Y., Lu, C.: An integrated job shop scheduling and assembly sequence planning approach for discrete manufacturing. J. Manuf. Syst. **61**, 27–44 (2021). https://doi.org/10.1016/j.jmsy.2021.08.003
26. Doh, H.-H., Yu, J.-M., Kwon, Y.-J., Shin, J.-H., Kim, H.-W., Nam, S.-H., Lee, D.-H.: Decision tree based scheduling for flexible job shops with multiple process plans. Int. J. Ind. Manuf. Eng. **8**(3), 621–627 (2014). https://doi.org/10.5281/zenodo.1091604
27. Yu, J.-M., Doh, H.-H., Kwon, Y.-J., Shin, J.-H., Kim, H.-W., Nam, S.-H., Lee, D.-H.: Decision tree based scheduling for static and dynamic flexible job shops with multiple process plans. J. Korean Soc. Precis. Eng. **32**(1), 25–37 (2015)
28. UniversalRobots: UR10e. https://www.universal-robots.com/products/ur10-robot/ [visited 01.08.2024] (2024)
29. Bosch: Intelligente Systeme für die wandelbare Fabrik 4.0. https://www.bosch.com/de/stories/industrie-4-0-flexible-production-line/ [visited 01.08.2024] (2024)
30. CleanAviation: Easy Come, Easy Go: Clean Sky's NextGen Cabin streamlines assembly and reconfiguration. https://www.clean-aviation.eu/clean-sky-2/results-stories [visited 28.03.2024] (2020)