How Does the Modeling Strategy Influence Design Optimization and the Automatic Generation of Parametric Geometry Variations?

Aritz Aranburu
Mondragon Unibertsitatea, Spain
aaranburug@mondragon.edu

Josu Cotillas
Mondragon Unibertsitatea, Spain
josu.cotillas@alumni.mondragon.edu

Daniel Justel
Mondragon Unibertsitatea, Spain
djustel@mondragon.edu

Manuel Contero
Universitat Politècnica de València, Spain
mcontero@upv.es

Jorge D. Camba
Purdue University, West Lafayette, IN, USA
jdorribo@purdue.edu

Abstract: The robustness and flexibility of a feature-based parametric CAD model determines the extent to which the geometry can be modified and reused in other design scenarios. The ability of a model to successfully adapt to changes depends on the type and sequence of the modeling operations selected to build the geometry, the parent-child dependencies defined during the modeling process, and the type and scope of the desired geometric change. Several formal modeling methodologies have been proposed to maximize model reusability, which have been shown to outperform unstructured approaches when designers need to manually modify the geometry. However, the effect of these parametric model strategies on the generation of valid solutions in heavily automated tasks has not yet been investigated. In this paper, we compare and analyze the performance of three well-established parametric modeling methodologies in various design optimization scenarios that involve the automatic generation of a large number of geometric variations. We discuss the results of a study with four parametric models of varying complexity and identify the limitations of each strategy in relation to the internal structure of the model. Our results show that explicit references and resilient modeling strategies are relatively robust for simple parts, but their effectiveness decreases significantly as the complexity of the model increases. In addition, we introduce the concept of intrinsic variability, which impacts the effectiveness of the methodology, and thus the quality of the parametric model, based on how the methodology is interpreted and executed.

Keywords

CAD Reusability, Parametric modeling, Modeling methodologies, Design intent, Design automation, Design Optimization, Robustness, Flexibility, CAD Quality

1. Introduction

Advances in computing and industrial technologies are transforming how design and manufacturing organizations operate and conduct their businesses. Modern Computer-Aided Design (CAD) tools, the exponential growth in our ability to capture data, and increasingly more powerful artificial intelligence algorithms are enabling the creation and simulation of sophisticated virtual product representations as well as the automation of many aspects of the engineering design process. The importance of the digital model in a product's lifecycle is evident through the variety of scenarios and activities where the model is used, such as analysis, Computer-Aided Manufacturing (CAM), and process planning, to name a few. It is particularly relevant in the context of a Model-Based Enterprise (MBE), where the virtual representation becomes the single source of truth and the central element around which all other activities revolve. The 3D model serves as the vehicle for managing, communicating, and sharing design information.

The authoring of digital product models for engineering typically involves the use of feature-based parametric CAD systems, usually history-based [1,2]. Parametric technology has become the industry standard paradigm in engineering design partly due to its ability to add semantics to the model and enable the rapid alteration and reuse of existing geometry. In an environment where the majority of design information is stored within the digital product model, CAD reusability enables design reusability [2], a key factor to reduce product development time [3]. The costs and implications of parametric modeling change in the broader context of engineering change management were recently discussed by Camba et al. [4].

Parametric feature-based models are built by combining high-semantic level geometric elements (i.e., features) in an associative hierarchical manner via parent/child relationships, which define a graph data structure of feature interdependencies that control the geometry. The specific definition of the feature dependencies and the internal structure of the model depend on the modeling process used to build the geometry, which is entirely the designer's responsibility [5]. In fact, it is not uncommon for two geometric models of the same part built by two different designers to have very different internal structures. When feature dependencies are defined effectively, any geometric change performed to a particular feature in the model will propagate automatically to all the features that depend on it, regenerating the geometry accordingly. When not defined properly, however, feature dependencies can be the source of many regeneration problems and drastically limit model reuse, even when the desired modifications are minor.

Theoretically, an unlimited number of approaches can be used to build a geometric model. From a purely geometrical point of view, any of these solutions is valid, as they all produce identical geometry and topology. However, the solutions may not necessarily produce comparatively reusable internal structures, and the models may react very differently to changes when modifications are made [2].

The vast majority of the strategies that can be applied to build a model, including the "trial and error" approaches that are often prevalent in industry [6], result in low-quality models that do not perform effectively in design reuse scenarios. Only a few strategies can maximize the flexibility and reusability of the parametric associative structure. Identifying the most appropriate modeling approach for a particular problem and devising the most effective way to organize the parametric dependencies are critical.

Throughout the years, some researchers and practitioners have attempted to formalize parametric modeling practices to maximize reusability, and various modeling methodologies have been proposed to ensure a certain level of consistency and standardization. Engineering organizations often develop internal modeling guidelines, which in many cases are not shared publicly for confidentiality and intellectual property reasons. Formal modeling methodologies have been shown to outperform non-structured approaches in terms of model robustness and flexibility to changes [1,2]. However, the benefits and impact of formal methodologies have almost exclusively been studied through the lens of manual design changes. For example, it has been shown that models created according to a formal modeling strategy, specifically the resilient modeling approach, can be modified by designers more quickly and easily than when other approaches are used [2]. Other similar studies have focused exclusively on parametric sketches [7–9].

In this paper, we examine the impact of the modeling methodology in design scenarios that involve automated processes. More specifically, we focus on design optimization tasks that require the generation of multiple design variations to find valid solutions to a problem. We present a series of experiments aimed at comparing three well-established modeling strategies used in feature-based parametric design (i.e., horizontal modeling, explicit reference modeling, and resilient modeling) by using four industrial CAD models with different levels of complexity in a design optimization study. To quantify their impact, we studied the behavior of the models when the geometry is modified, emphasizing the correctness of the models after changes, the number of successful variations generated with respect to unsuccessful variations, and the total processing time required to produce the solutions.

2. Background

2.1 CAD model complexity metrics

In order to determine the quality and reusability of a parametric 3D model, it is critical to measure its complexity [10–12]. However, there is no consensus on a formal definition of 3D model complexity [6,13,14]. The concept is multifaceted with many interrelated dimensions. Some of the dimensions that have been proposed include, for example, the coupling between parts in an assembly and the exclusive specifications of assembly joints [15,16], the effort required to manufacture or design the part [17], the number of features and entities in each sketched feature [18–20], aspects related to the complexity of the design process [11], size, interconnectivity, and decomposition of the corresponding connectivity graph [21], the number of sketched and edge features [22], dimensional constraints [23], faces [1], and the number of each unique paths between each pair of nodes [24], the degree to which nodes are grouped within the system to measure interconnectedness between features [25], the number of independent paths through the graph (Cyclomatic complexity) [26], the string length when the graph is encoded as a binary string (Kolmogorov complexity) [27], and the graph entropy measure, which describes the uncertainty of a system using the algorithm of Li et al. [28].

The previous list is not exhaustive. Authors Camba et al. [2] compiled and analyzed various metrics that can be used to quantify the complexity of a 3D CAD model. In their study, the authors concluded that the number of faces, edges and vertices are reliable indicators of geometric complexity, but other metrics such as the number of features, the number of dependencies, and the number of leaf nodes and average node connectivity in the corresponding graph are needed to describe the complexity of the structure of a parametric model.

Amadori et al. [6] claimed that flexibility is an important aspect of model quality, which is related to the robustness and size of the design space of the model. The flexibility of a geometric model refers to the ability to represent a wide range of different product configurations, layouts, and sizes. The wider the range of geometries the model can create, the greater the flexibility. Robustness refers to the errors or instability problems that can be caused by changes in the geometric model. Fewer errors lead to greater robustness. The size of the design space is determined by the designer, who decides the range of input variables. It is crucial to define all input variables from the start, as these variables will affect the future behavior of the model. By defining a concise/limited design space, designers can create models that are robust and flexible.

The adaptive nature of the design tree allows designers to model complex parts quickly and with relative ease, while increasing the flexibility and reusability of their designs. In order to fully leverage the reusability benefits enabled by parametric feature-based CAD, models must effectively react to the most likely geometric variations. When feature dependencies are defined efficiently, alterations made to a parent node will automatically propagate to its child nodes, and the CAD model will react to changes in a predictable manner [2].

Unfortunately, parent/child feature interdependencies are at the root of many regeneration problems in parametric modeling. The size and complexity of a parametric CAD model can grow rapidly depending on the scenario. As the number of dependencies grows, so does the interconnectedness of the design tree, which can negatively affect the maintainability and reusability of the model [2]. When feature interdependencies are not properly defined, even minor alterations can cause the CAD model to become unstable, forcing designers to rebuild part of or the entire model to restore the intent of the new design [2].

While many options exist to generate a geometric solution, the robustness and reusability of the model depends greatly on the experience of the designer and the methodology. The “desirable range” is the optimal point at which the designer achieves a robust and flexible parameterized model that can accommodate a wide range of variations, and effectively communicates design intent [29]. The concept is illustrated in Fig. 1.

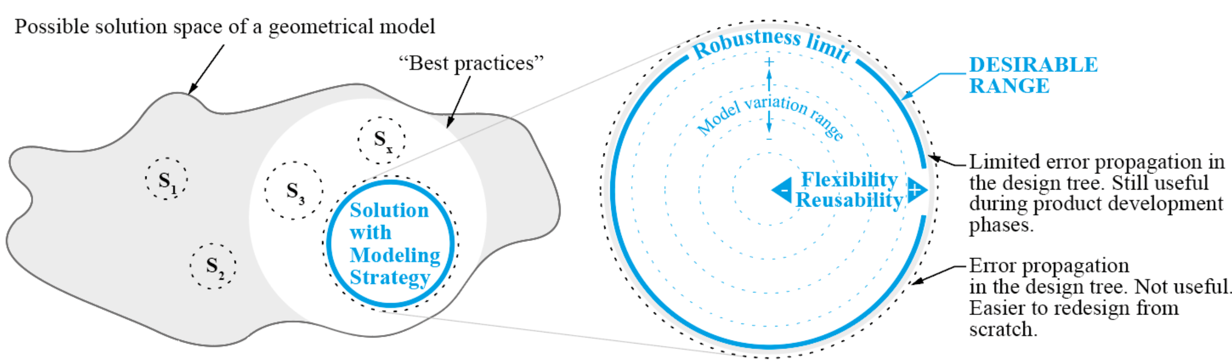


Fig. 1. Solution space identifying the desirable range where the optimum relationship between flexibility and reusability and the variation range of the model is achieved [29] (adapted from [1, 6])

2.2 Formal modeling methodologies

Various formal modeling methodologies have been proposed to maximize model quality, robustness, and flexibility against changes, most notably horizontal modeling [30], explicit references modeling [1], and resilient modeling [31]. In this section, we review the basic principles of each of these strategies.

2.2.1 Horizontal Modeling

The Horizontal Modeling Methodology is a modeling strategy patented by Delphi Technologies [30]. The goal of the methodology is to prevent propagation problems resulting from design changes by eliminating all dependencies between the elements in the design tree. Landers and Khurana [30] stated that the problems caused by parent/child relationships originate from the classic vertical tree structure. They proposed minimizing or eliminating interdependencies by ensuring all features are generated independently from the others and always referenced to datum planes. Through these datum planes, features are positioned at the same level within the tree, creating the horizontal structure. The concept is illustrated in Figure 2. This scheme minimizes the potential errors caused by modifications to a feature, reducing regeneration errors.

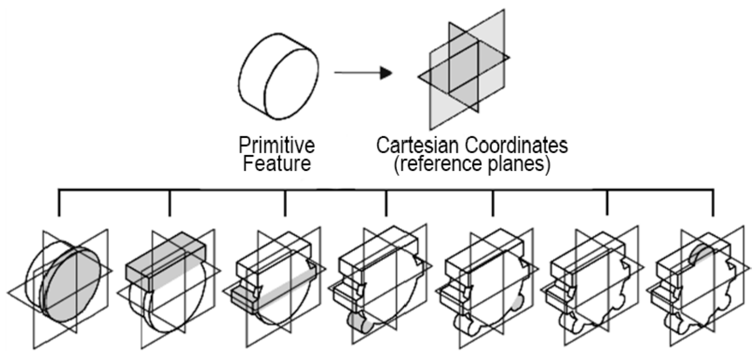


Figure 2. Horizontal Modeling process (adapted from [30]).

In addition to the "horizontalization" of the tree structure, the construction of horizontal models must also consider how the part will be manufactured. All features that comprise the main body are modeled first, followed by subtraction features.

The creation of datum planes adds an additional level of complexity and reduces parameterization. The chain of feature dependencies is usually short and easy to trace, but systematically eliminates parent-child relationships. The horizontal methodology has been criticized for making it difficult to convey design intent in the feature tree, and for the eliminating the automatic propagation of geometric changes, a functionality that is at the core of the feature-based parametric paradigm. In addition, there are no specific guidelines on how to organize or name the elements in the design tree.

2.2.2 Explicit Modeling Methodology

The Explicit Reference Modeling (ERM) methodology was proposed by Bodein et al. [1] to address the second level of their CAD Efficiency Roadmap [32]. The authors proposed minimizing the number of possible solutions to generate a model (i.e., standardize model creation) by minimizing the constraints or relationships associated with existing geometries.

To apply this methodology, the number of parent/child dependencies must be reduced as much as possible without "un-parameterizing" the associativity of the model (as in horizontal modeling). To this end, the authors distinguish between two types of constraints:

- Category 1 associations/constraints: Associations that can avoid being related to existing geometries. To avoid references to existing geometries, explicit reference entities such as planes or lines must be generated. Reference entities can be created as explicit references using elementary parametric elements such as points instead of vertices, or planes or surfaces instead of existing faces.
- Category 2 associations/restrictions: Associations to existing geometry are mandatory. The goal in this category is to reduce parent/child relationships as much as possible to minimize dependencies between operations. Consequently, operations such as rounding, chamfering or hollowing/shelling should be applied as close as possible to their original primitives in order to reduce the degrees of feature dependency [1], as illustrated in Figure 3.

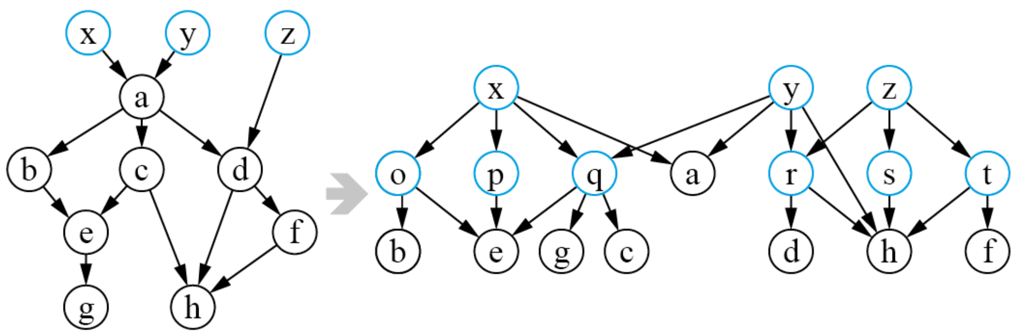


Figure 3. Example of graph structure in a parametric model: unstructured methodology (left) vs. ERM (right) [1].

Constrained features in Category 2 can cause models to become unstable or cause problems if modified or removed due to the parent/child relationships. The goal is to reduce feature dependencies by placing operations with category 2 constraints as close as possible to their parent or primitive operations.

Bodein et al. [1] also proposed the implementation of a functional approach in the methodology, which focuses on the identification of the functional aspects of the part so they can be addressed independently. The approach consists in four steps: (i) functional breakdown of the geometry, (ii) creation of explicit references for each functional part, (iii) creation of the necessary solids for each functional part, and (iv) linking the functional bodies with Boolean operations. The strategy concentrates error propagation on one functional part, in case errors occur.

2.2.3 Resilient Modeling Strategy

The resilient modeling methodology (RMS) [34] was proposed by Gebhard [31] and focuses on creating robust and reusable models by optimizing parent-child dependencies and structuring easily understandable design trees. The methodology is based on four main principles:

- Renaming elements (so that they are intuitive) to improve communication.

- Tree structure organized into eight groups.
- Categorization and associations of the features according to their volatility.
- Stress testing by modifying the critical parameters and verifying the behavior of the model.

These four principles ensure that the communication of design intent is effective and easy to understand by third party designers. By organizing the tree in the same manner, access to the information is consistent in all models. The structuring of the design tree into eight groups describes what type of operations should be part of each group and what rules should be followed when creating each group. The eight groups are summarized in Table 1.

Table 1. Feature groups defined in the resilient modeling strategy [34].

Group	Description	Typical features	Notes	Links
1. Skeleton	Is the specification sheet for the solid model	· Ref Bodies, Layouts, Sketches, Ref Planes, Coord. Sys, Images, Surfaces, Project, Extend, 3D Curves, Trim, Split	· No solid features · No duplicate dimensions	Allowed
2. Core	It captures the model's basic shape, size and orientation	· The first group to contain solid features · Only profile-based features: Extrude, Sweep, Thin Wall, Revolve, Loft, Shell · Shell feature as an exception	· No local profiles · No numeric extents · Add material	Allowed
3. Surface	It contains surface or curve features that would normally be in the skeleton group, but they must be linked to the Core Group	· No solid features: Surfaces, Project, Extend, 3D Curves, Trim, Split · Must depend on the "2. Core Group"	· Surface group feature are used as boolean in the 4. Detail Group · A "final feature" in the modify group · A "deferred feature"	Allowed
4. Detail	Features are for add definition to a model using local geometry	· No reference features · No links to other "Detail" features · Extrude, Sweep, Hole Revolve, Loft, Thread	· Use hole features instead of a cut · Detail features can me moved or suppressed without causing an error	No link between detail features
5. Holes	Are a cylindrical cavity created by an extrude cut, revolved cut or a hole feature	· No reference features · No links to other "Hole" features	· Use hole features instead of a cut · Those features can be moved within the group or suppressed without causing an error	No link between hole features
6. Modify	Features that replicate or transform groups of features	· First Mirror, Pattern, etc. · Followed by transform features: Draft, Pattern, Mirror, Final Features ·Final features are last: add or remove	· Don't pattern Core features · Suppressible Draft features material from the modified model	Allowed
7. Quarantine	Those features consume their defining hard edge, replacing them with derived edges	· Don't consume a defining face · Chamfer, Blend, Round	· Chambers are first. · Rounds & Fillets are next · Use caution when linking to other quarantine features · If creation error occurs, change the feature order · Split the features	Allowed
8. Variant	Used to generate versions of the base part by suppressing or unsuppressing the group	·Multiple groups are allowed	· Direct editing features are common in this group	Allowed

2.3. Design optimization and automatic generation of geometric variations

The responsiveness of 3D models to geometric variations in a stable manner is of considerable relevance in automated design tasks [35]. This is particularly the case in design tasks where multiple design variations are required to explore, optimize, or generate new designs. The most common of these tasks in industry include [35] i) design optimization, ii) simulation, iii) product/part families and configurations, and iv) generative design algorithms.

i) Design optimization tools facilitate the alteration of a CAD model based on user input. By specifying a set of criteria, engineering constraints and the design objective to be achieved, geometric variations from a source 3D model are automatically generated. When parameters and dimensions of the 3D model are changed, the tool evaluates different design scenarios and optimizes the design by selecting the best scenario from a number of possible combinations. To explore all possible design scenarios the models must be efficiently parametrized, robust, and flexible. If not, the optimization will run unsuccessfully (due to regeneration errors), or even worse, potentially valuable design alternatives may be ruled out.

ii) Simulation tools assist designers to assess key design and manufacturing factors by evaluating and predicting the performance and behavior of parts/products based on physical laws. Simulating 3D models, designers can verify whether the design requirements are converging into the objective, which delimits the part/product design space to explore in each iteration [36]. We say iteration since each simulation (unless it is the last one) results in a minor or major design change. Such changes frequently require alterations to the geometry of the model, particularly when these tools are used in the early stages of the design process.

Inefficient modeling practices can significantly contribute to costly geometrical changes. In a recent study, Nerenst et al. [37] identified the lack of robust CAD as a critical barrier for effectively completing simulation tasks. The use of low-quality models often leads to issues during simulation stages. Indeed, it is estimated that roughly half of all engineers spend over four hours a week fixing design data, and 15% spend more than 24 hours a week on the same design activity [38]. The complexity of engineered products is increasing, and the most up-to-date software can analyze the overall performance of a product simulating all influences simultaneously. As a result, simulation is used to analyze thousands of possible alternatives until the optimal design is identified. In this context, to meet the demands of geometrical variation the digital model must be flexible and robust.

iii) Parametric model configurations are derived versions of an original part/product made after the first model has been validated. It is necessary to identify the parameters and features that will change and assign values for each scenario. This quickly establishes a family of similar parts or assemblies responding to new needs or other cases of application. Those configurations can be defined and controlled internally within the geometry of the original model, or externally, through a spreadsheet-like data grid stored as a separate file. As in previous cases, the successful generation of configurations mainly depends on the robustness of the parametric model. If the model is not robust enough, it will need to be extensively edited and/or built from scratch.

iv) Generative design techniques assist in the creative process in a more integrated manner [39] by helping the user to make design decisions. The approach involves the use of algorithms to automatically construct optimal geometry [40]. As an example, the algorithm of Krish [41] is able to explore design options in the conceptual stage without human intervention. However, the nature of parametrization tasks can make a big design space problematic to explore. Two approaches exist to address this problem: the first is a strategy proposed by Khan and Awan [40] where important features are first parametrized with a large number of geometric parameters. After some iterations, problematic parameters which might be or are less relevant to the overall variation, are eliminated. The second strategy proposed by Khan et al. [36] is intended for the conceptual stage and is focused on capturing user preferences using geometric constraints and reducing the design space in each iteration.

Finally, if a cost-effective strategy can be defined to parametrize detailed and robust 3D parametric models, models can be built to deliberately address geometric variations and be used as the basis for generative design techniques. Such algorithms could contribute beyond the conceptualization phase and have applications in sectors such as naval (e.g., the parametrization of the hull geometry [42]) and railway (e.g., the aerodynamic design of the heads of high-speed trains [43]).

In this paper, we examine how the modeling strategy impacts the robustness and flexibility of a parametric model when the geometry needs to be modified for tasks that involved the iterative generation of design variations, such as the scenarios described above.

3. Experimental procedure

To determine how the modeling methodology influences the generation of valid geometric variations, we conducted a series of tests with four representative parametric models of varying complexity. All tests were completed in a computer laboratory environment equipped with a workstation (CPU, Intel Xenon 2295 @ 3.00GHz; RAM, 64,00 GB Dual-Channel @ 1462MHz; and Graphic card, 4083MB NVIDIA RTX A5000) and a commercial parametric associative CAD system (i.e. DS SolidWorks). In order to generate geometric variations automatically and evaluate the success of specific design scenarios, we leveraged the tool Design Studies, which is fully integrated within the SolidWorks environment. Similar tools are available in other CAD packages (e.g., Design Studies in Creo Parametric, or Geometry Optimization in Siemens NX). The tool iterates through all possible geometric combinations of a particular model within a design space based on varying geometric parameters within a particular range, which are predefined by the designer.

In our study, Design Studies was used to evaluate the robustness of a series of models built according to three formal parametric modeling methodologies. For each model, we defined a series of dimensional changes and compared the percentage of design scenarios that were successfully regenerated for each methodology. By analyzing how a model responds to changes as well as the processing time required to regenerate the parametric geometry, we can determine robustness as well as identify key factors in the construction of the model.

3.1 Sample

Four representative industrial parts were selected to test and compare the modeling methodologies: (A) a connecting rod, the part of a piston engine which connects the piston to the crankshaft; (B) a pump housing, the part where the pump is located and through which the liquid is directed; (C) a steering knuckle, the component which contains a wheel hub and is connected to the suspension and steering components; and (D) a bell housing, the part of a transmission that covers the flywheel and the clutch. The parts were intentionally selected to reflect increasing levels of geometric complexity according to various metrics. The level of detail of each part is based on its functional requirements and the number of connecting or interacting parts in their respective sub-assemblies. The parts are depicted in Fig. 4.

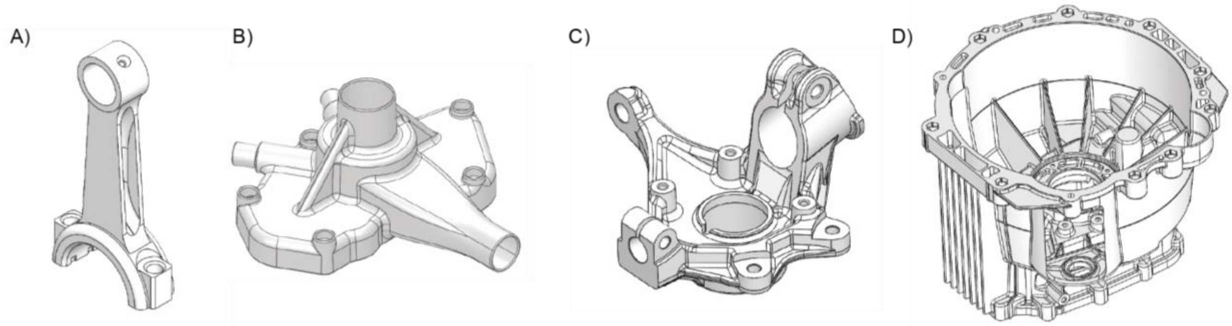


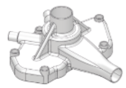
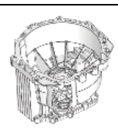
Fig. 4. Parts used in our study: connecting rod (A), pump housing (B), steering knuckle (C), and bell housing (D).

3.2 Methodologies and model complexity metrics

For each part depicted in Fig. 4, four different feature-based CAD models were created, each according to the guidelines of the different methodologies discussed in Section 2. Although the objective of a formal modeling methodology is to ensure consistency and obtain quality models regardless of the level of expertise of the designer, the structure of the model may depend on how the methodology is interpreted and executed during the modeling process (i.e., intrinsic variability). In our case, all the models were created manually by expert engineers with significant industry experience and extensive knowledge of the various methodologies. In addition, after each model was created, it was carefully reviewed by other design engineers to verify that the methodology was applied correctly and rigorously.

The characteristics of each model are summarized in Table 2, including geometric properties (number of faces, edges, and vertices) [2], the three basic metrics to assess parametric features and dependencies (number of features, dependencies and average node connectivity) [2], Li entropy (the uncertainty of the graph) [28], Kolmogorov complexity (string length when the graph is encoded as a binary string) [27] and Cyclomatic complexity (the number of independent paths through the graph) [26].

Table 2. Complexity of the models used in our study

Part		Version	Metrics								
			No. of Faces	**No. of Edges**	**No. of Vertices**	**No. of features**	**No. of dependencies**	**Average node connectivity**	**Li entropy**	**Kolmogorov complexity**	**Cyclomatic complexity**
A		Horizontal v1	78	203	124	36	59	2.66	171.84	714	40
		Horizontal v2	78	203	124	31	50	2.62	140.51	505	34
		Explicit	78	203	124	39	101	2.47	266.01	1218	73
		Resilient	78	203	124	37	95	2.26	246.98	1146	69
B		Horizontal	384	806	414	52	89	2.96	281.03	1074	66
		Explicit	384	806	414	52	119	2.83	350.33	1434	90
		Resilient v1	410	872	454	49	101	2.66	300.61	1218	75
		Resilient v2	416	883	459	48	102	2.62	299.71	1230	76
C		Explicit	622	1,481	851	277	731	3.56	2,976.75	13,167	574
		Simpl. Explicit	195	506	303	188	533	3.32	1,992.14	8,536	404
		Resilient	646	1530	877	249	731	3.28	2,863.54	11,704	567
		Simpl. Resilient	195	516	314	179	547	3.07	1,995.02	8,760	410
D		Explicit	3,658	8,275	4,617	327	640	3.74	2,694.28	11,529	469
		Simpl. Explicit	853	2,329	1,485	177	429	3.54	1,633.40	6,872	311
		Resilient	3,603	8,130	4,530	361	779	3.92	3,435.62	14,139	566
		Simpl. Resilient	849	2,311	1,473	221	484	3.61	1,956.79	7,752	328

Each part in our study was modeled four times, instead of three, because a particular modeling method could be interpreted in two different ways, resulting in differences that were deemed significant. For example, for Part A, two horizontal models were created (v1 and v2), both of which fully comply with the guidelines of the methodology. However, differences in how the sketches are created and related to other elements may result in differences in the respective design trees and thus in the resulting solution spaces, as shown in Fig. 5. Likewise, two resilient modeling approaches were considered for part B. In this case, the decision was based on the construction and placement of the cavity within the design tree. In the first version of the resilient model (v1 in Fig.6), the shell feature (called “M2-Shell 2 mm”) used to create the cavity is located in the “modify” group. However, it is also possible to place this feature in the “core” group (as shown in v2 in Fig. 6). Since no explicit guidelines are defined regarding this scenario, a decision was made to consider both cases.

We define the term *intrinsic variability* of a modeling methodology as the inherent variances in the structure of a parametric model that may result from differences in how the methodology is interpreted and executed during the modeling process. The intrinsic variability of a methodology can be significant as well as impact its effectiveness. From a quality standpoint, methodologies with low intrinsic variability yield more consistent results.

No horizontal models were created for parts C and D. The intrinsic complexity of these parts requires the creation of many datum planes, even before creating solid bodies, which quickly becomes unmanageable and impractical. In addition, the lack of parameterization enabled by the elimination of parent–child relationships, results in design changes that do not propagate automatically even when the geometry regenerates successfully. This situation is particularly problematic as errors may easily go unnoticed, making the model appear robust even when the design intent is not maintained. Instead, we constructed additional versions of the explicit and resilient models by simplifying details such as chamfers, fillets, and other features that are highly dependent on the main geometry but are located towards the bottom of the design tree.

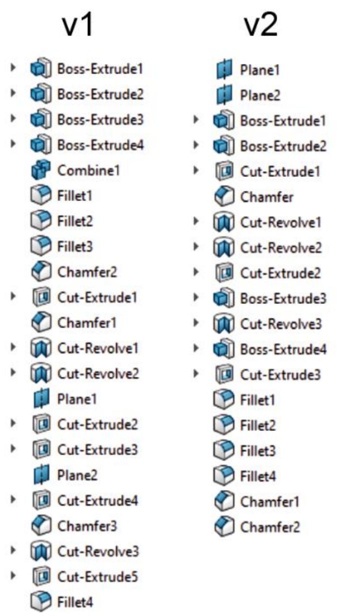


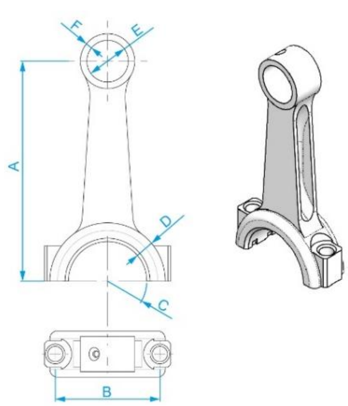
Fig. 5. Design trees of horizontal models v1 and v2 for Part A (connecting rod).

Fig.6. Design trees of resilient models v1 and v2 for part B (pump housing).

3.3 Solution space constraints

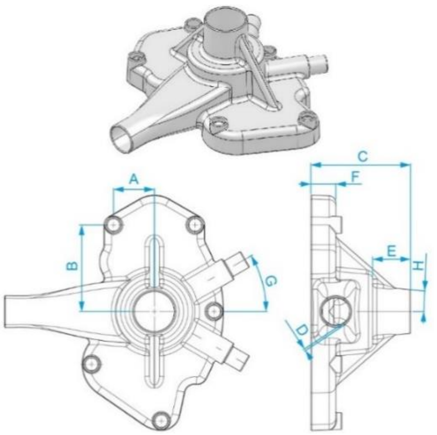
To limit the design space for the parts in our study, we identified a set of functional dimensions that were most likely to change for these types of parts in an industrial setting. The dimensions and the respective ranges of variation for all the parts are shown in Tables 3-6, respectively.

Table 3. Functional variables and range of values for Part A (connecting rod).



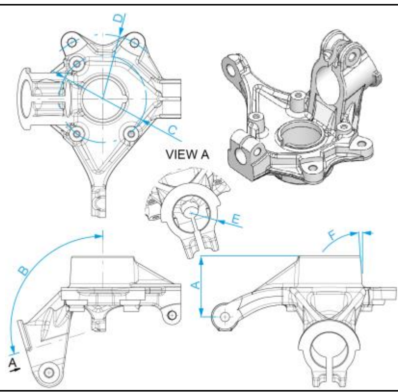
Parameter	Minimum value (mm)	Maximum value (mm)	Step (mm)	Scenarios per parameter
A-Dist. between axis	80	140	20	4
B-Dist. between bolts	22	26	2	3
C-Radius axis A	18	24	2	4
D-Thickness axis A	7,5	9,5	1	3
E-Diameter axis B	16	22	2	4
F-Thickness axis B	2	4	1	3
			Total scenarios	1,728
			Solution space to explore	1,729

Table 4. Functional variables and range of values for Part B (pump house).



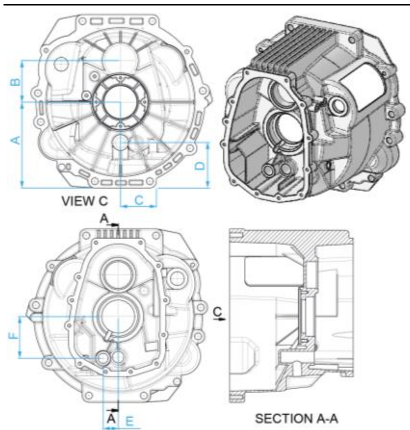
Parameter	Minimum value (mm)	Maximum value (mm)	Step (mm)	Scenarios per parameter
A-Bolt hole Z	87	89	1	3
B-Bolt hole X	41	43	1	3
C-Main axis length	80	140	20	4
D-Thickness	2	3	1	3
E-Stepped axis length	38	46	4	3
F-Base height	25	50	12.5	3
G-Small axis angle	30	45	15	2
H-Draft angle	1.5 deg.	4.5 deg.	1.5 deg.	3
			Total scenarios	3,888
			Solution space to explore	3,889

Table 5. Functional variables and range of values for Part C (steering knuckle).



Parameter	Minimum value (mm)	Maximum value (mm)	Step (mm)	Scenarios per parameter
A-Lateral Flange Y	19,03	31,03	4	4
B-Long Axis Angle	104.23	109.23	2.5	3
C-Inner Bolts Diam.	105.02	109.02	2	3
D-Outer Bolts Rad.	76.77	82.77	2	4
E-Long Axis Rad.	31	33	1	3
F-Draft Ang.	4	6	1	3
			Total scenarios	1,296
			Solution space to explore	1,297

Table 6. Functional variables and range of values for Part D (bell housing)



Parameter	Minimum value (mm)	Maximum value (mm)	Step (mm)	Scenarios per parameter
A-Main axis height	187	189	1	3
B-Dist. b/w main axis	88	91	1	4
C-Aux. axis1- Y	76.98	78.98	1	3
D-Aux. axis1 - Z	87	89	1	3
E-Aux. axis2 - Y	31.5	33.5	1	3
F-Aux axis 2 - Z	89.75	91.75	1	3
			Total scenarios	972
			Solution space to explore	973

4. Results

The results of our experiment are shown in Table 7.

Table 7. Experimental results for each part.

Part	Methodology	Total scenarios (T)	Successful scenarios (S)	Success rate ($S/T \cdot 100$)	Total scenarios analyzed for Design Intent (Z)	Design Intent is maintained (X)	Design Intent Success rate ($X/Z \cdot 100$)	Regeneration time (s)
A	Horizontal v1	1,729	49	2.83%	49	1	0.08%	0.17
	Horizontal v2	1,729	37	2.14%	37	1	0.08%	0.27
	Explicit	1,729	1,297	75.01%	297	297	100%	0.20
	Resilient	1,729	1,297	75.01%	297	297	100%	0.37
B	Horizontal	3,889	100	2.57%	100	12	0,003%	1.27
	Explicit	3,889	1,189	30.57%	291	241	82.82%	1.53
	Resilient v1	3,889	262	7.73%	262	200	76.34%	3.12
	Resilient v2	3,889	197	5.07%	197	155	78.68%	4.37
C	Explicit	1,297	15	1.15%	15	15	100%	6.06
	Resilient	1,297	1	0.08%	1	1	100%	6.54
	Explicit (simpl.)	1,297	1,297	100.00%	297	0*	0%	1.74
	Resilient (simpl.)	1,297	866	66.77%	266	266*	100%	2.37
D	Explicit	973	33	3.39%	33	33	100%	16.96
	Resilient	973	30	3.08%	30	30	100%	17.32
	Explicit (simpl.)	973	973	100%	276	276*	100%	2.99
	Resilient (simpl.)	973	972	99.90%	276	276*	100%	4.45

* NOTE: design intent results for simplified versions of Parts C and D are not comparable.

In our study, an error is defined as any event that fails to regenerate the geometry correctly, i.e. a failure in one or more features in the model, as reported by the system. It is important to note that a priori we do not know whether the geometric changes performed to a model will be feasible or not. However, these cases do not affect our study, since we are conducting a relative comparison of the performance of the methodologies, and not an absolute comparison of the corresponding design spaces. If a particular change is not geometrically realizable, it will not be realizable in any of the methodologies, regardless of the modeling approach.

We understand design intent as a CAD model's anticipated behavior when altered, as discussed by Otey et al. [44]. The preservation of design intent for the successful cases was verified manually, as this dimension of CAD quality is extremely difficult to check automatically and even represent in an explicit manner [46]. For models where the number of successful scenarios was more than 300, a simple random sampling strategy was applied [45], as follows:

$$n = \frac{N \cdot Z^2 \cdot p \cdot q}{d^2 \cdot (N-1) \cdot Z^2 \cdot p \cdot q} \tag{1}$$

Where:

- n = sample size
- N = total number of scenarios
- Z = confidence level (95%)
- d = precision (5%)
- p and q = estimated proportion (0.5)

The preservation of design intent was determined by comparing the resulting scenarios (Z), i.e., the modified models, with the original (unmodified) one. For the comparisons, we considered whether topological inconsistencies were present in any of the functional elements of each part,

such as the incorrect removal of features (e.g., ribs, holes, fillets, etc.).
We note that the simplified models for parts C and D are not comparable in terms of design intent. These models are simplified versions of the final geometries for each methodology and not intermediate models at an earlier stage of construction. More specifically, they are models without fillets. In the resilient modeling strategy, the removal of these features is not critical, as fillets are created at the end of the process, and thus they are always at the bottom of the model tree (they have no child features). However, in the explicit references modeling strategy, the removal of fillets can affect the parent-child structure of the features as well as the sequence of operations, which can affect the preservation of design intent.

For Part A (connecting rod), a total of 49 variations were generated with no errors for the horizontal model v1, but design intent was not maintained. In the case of horizontal model v2, 37 variations were generated successfully, but design intent was also lost. For all successful scenarios in both models, design intent had to be verified manually, as many geometric problems were difficult to identify. Only two scenarios (one in v1 and one in v2) maintained design intent. Some examples that illustrate loss of design intent in these models are shown in Fig.7.

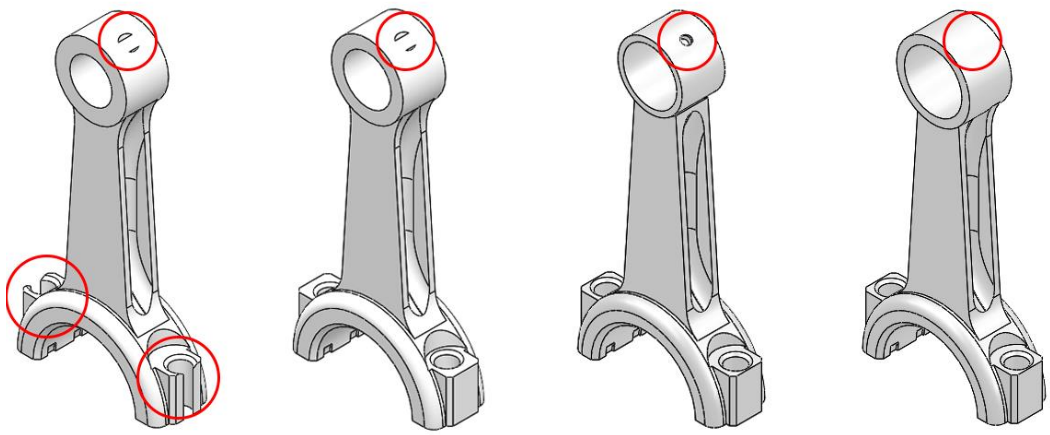


Fig. 7. Geometric variations generated successfully but with loss of design intent (circled in red).

The explicit methodology produced 1,297 successful geometric variations out 1,729, which is 75.01% of the design space. For the variations that failed to regenerate, we identified a sketch in one of the cutting features that flipped direction unexpectedly for certain values of one of the dimensions, as illustrated in Fig. 8.

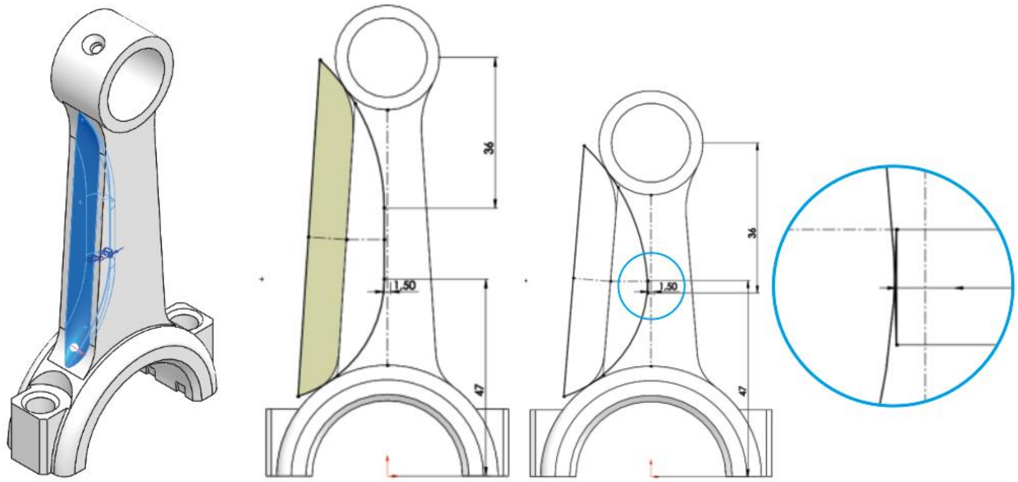


Fig. 8. Source of errors in the explicit model of the connecting rod

For Part B (pump house), the horizontal modeling approach was the least effective with 100 geometric variations generated successfully in a design space of 3,889 (2.57% success rate). Furthermore, design intent was only maintained in 12 of these cases (verified manually), which were generated based on three parameters: thickness, small axis angle, and stepped axis length. In all cases where design intent was not maintained, there was error propagation within the same feature.

The explicit modeling method for part B resulted in 1,189 successful scenarios out of 3,889 (30.57% success rate), and 82.86% success rate maintaining the Design Intent, which was the highest. The first version of the resilient model resulted in 262 successful scenarios (8.82% success rate and 76.34% maintained the Design Intent), whereas the second version produced 197 successful scenarios (5.07% success rate and 78.68% maintained the Design Intent). For this particular part, a significant number of errors were caused by the regeneration of cosmetic features, such as fillets, as certain changes to the main body of the part resulted in differences in the edges in which fillets are referenced.

Fillets were also problematic in Part C (steering knuckle), as the explicit and resilient methodologies require fillets—and to a lesser extent chamfers—to be modeled differently. These differences are significant because approximately 70 features in the steering knuckle (~15% of the total) are fillets. Indeed, in models where multiple fillets intersect, overlap, and/or meet at a vertex, it is virtually impossible to construct the exact same geometry with both methodologies. For this reason, both explicit and the resilient models of part C were analyzed with and without fillet features.

For the models with fillets, the explicit methodology yielded the highest success rate (1.15%) with 15 out of 1,297 successful scenarios (100% maintained the Design Intent), whereas only one successful scenario was generated with the resilient strategy. For the version of the models without fillets (simplified), the explicit methodology once again yielded the best results with 1,297/1,297 successful scenarios (100% success rate). The resilient model yielded 866/1,297 successful scenarios (66.77% success rate). It is clear that fillets are at the root of a significant number of unsuccessful scenarios. In the simplified version of the explicit model, we considered that design intent was not maintained in any scenario because two ribs could not be rebuilt when the fillets were removed. In resilient modeling, however, fillets are defined at the bottom of the design tree (and thus have no child features), which allows the successful regeneration of these ribs when fillets are suppressed, preserving design intent.

Finally, the Part D (bell housing) results show that the explicit methodology is slightly more robust than resilient modeling, with the former giving rise to 33 successful cases out of 973 (3.39%), and the latter 30 successful cases out of 973 (3.08%). In the versions where the rounding features are removed, the models achieved robustness of 100%, or close to it. However, this result must be understood in the context that the models have around 100 fillet and chamfer features, which means that 34% percent of the operations were eliminated. Thus, it is evident that the tested methodologies lose their effectiveness when presented with models with numerous operations and increased numbers of fillets and chamfers. Design intent is maintained in all scenarios, including the explicit modeling methodology. Unlike the explicit model for Part C, where the removal of fillets caused problems in the regeneration of the ribs, design intent in maintained in part D as the removal of fillets does not involve a functional change in the part.

5. Discussion

The results of our study show that for the first part (the connecting rod) comparable levels of robustness can be achieved by both explicit and resilient modeling methodologies and that the horizontal modeling strategy is inefficient and significantly underperforms when compared to the other two. However, the structure of the resilient model is less complex, as shown by the complexity metrics of the model: reductions of 9% in Li entropy; 8% in the number of dependencies, average node connectivity, and Kolmogorov complexity; 7% in cyclomatic complexity; and 5% in the number of features. However, the regeneration time for the explicit models is 45.95% less in part A.

Our results show that resilient modeling is a robust methodology and models are the most reusable in the face of manual changes, as confirmed by other authors [2]. However, in automated processes where large numbers of models need to be produced without human intervention, the explicit references modeling methodology yields comparable results in terms of robustness but has faster regeneration times, making the automated processes more efficient.

For more complex parts (parts B and C), explicit models once again have more complex internal structures than their resilient counterparts, but significantly higher levels of robustness are achieved (4 to 6 times better than resilient models for part B, and 435 times better than resilient models for part C). For part B, the regeneration time of the explicit models is between 50.96% and 65% less than resilient models. In the case of part C, the differences in regeneration times are not so pronounced (7.34% less in the explicit model than the resilient model). For part C, both the simplified explicit and resilient models have very high levels of robustness (100% for explicit and 66.67% for resilient), but the regeneration time is lower in the explicit models (32.3% less than in resilient).

Our results could initially suggest that working with resilient models could be more agile, since disabling all fillets and making significant changes to the model should be easier. However, as in explicit models, fillets may not always be directly associated with the geometric features that are going to change. Therefore, the behavior of geometrical changes is unexpected due to modifications that are not propagated to directly related dependent features. In resilient models, the geometry is developed and finished in edges. While in explicit models, the geometry is developed progressively including fillets. Therefore, the initial topology to apply the fillets is completely different in certain areas that have required a considerable group of operations to create a geometry. The different strategies used to manage fillets in the two methodologies makes it impossible to generate exactly the same geometry as illustrated in Fig.9. In the figure, the SolidWorks function “Body Compare” was used to analyze two topologies of the models of Part C. This tool is used to compare CAD models (B-Rep or parametric models) in reverse engineering activities or to compare CAD models with manufactured and scanned parts for deviation analysis in quality tests. The deviations are calculated by selecting a body to compare and a source body. The differences between selected bodies are shown by a color gradient that represents where they match (or not) and how much the differences are.

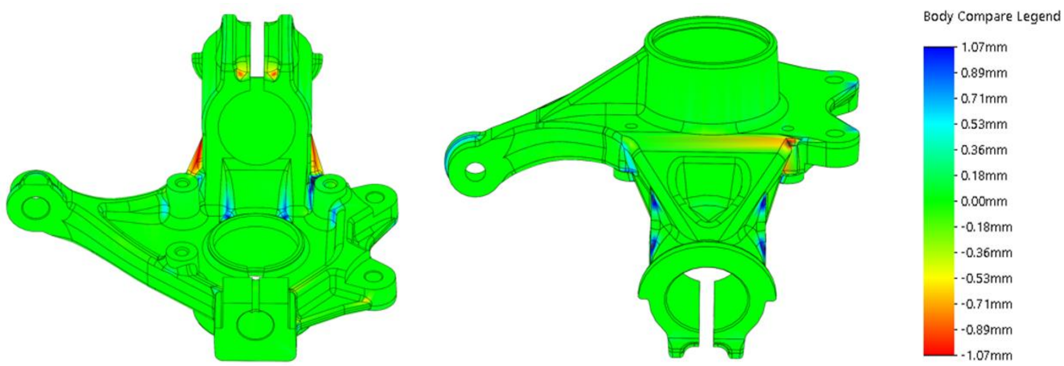


Fig.9. Geometric comparison analysis between explicit (left) and resilient (right) models of part C.

For part D, the level of robustness and the regeneration times are similar for the explicit and resilient models. In contrast to part C, the fillets and rounds in the models for part D do not affect the results. However, during modeling, the resilient strategy was found to be the least effective solution for model four of part D, as the intrinsic limitations of the methodology made it difficult to place the various operations in the proper categories. Specifically, the resilient methodology defines seven folders to organize the features, but the construction of this particular model does not begin with a single main body in the core folder. In this case, the part is generated by several bodies that in parallel must subsequently be associated with other features without being related to each other, as shown in Fig.10. The methodology requires a significant amount of planning because the possible routes for creating new features are greatly reduced as the modeling process progresses. In particular, the large number of dependencies within the shelled features, the various Boolean operations, and the ribs made the model structure difficult to manage, as many operations had to be grouped in the “modify” folder. As mentioned earlier, the independent bodies, which are modeled first, are later joined and completely defined in the modify layer, which is similar to what happens in the explicit methodology. However, the resilient modeling process becomes much more complicated because the design tree is structured by layers, instead of functional bodies, which creates uncertainty as there is not a well-defined procedure that describes how to operate in such cases.

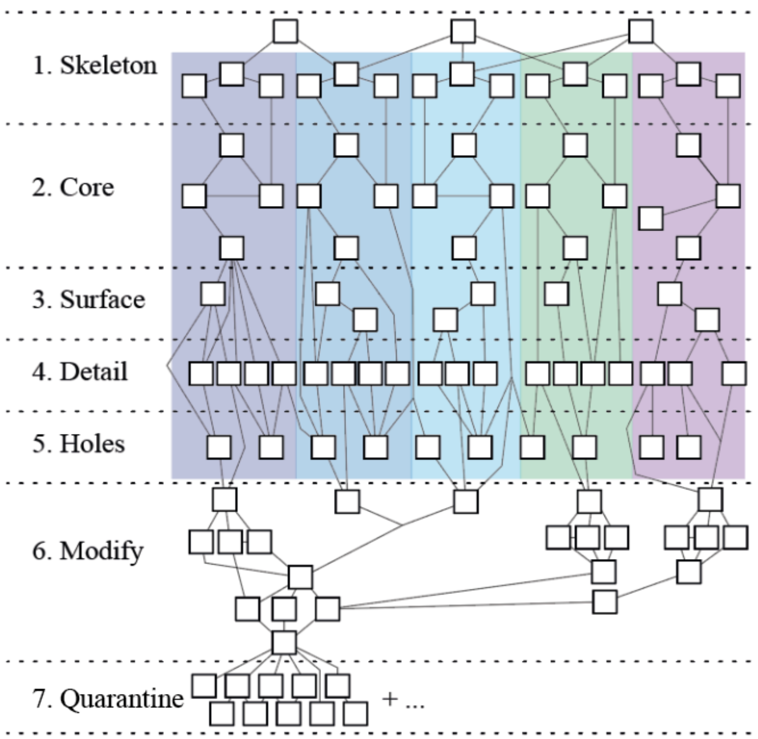


Fig. 10. Simplified associative structure of the resilient model of Part D. Each square represents a feature or a group of features, the lines are dependencies, the rows are folders in the design tree of the Resilient Methodology, and the colored areas represent different bodies.

The explicit modeling methodology allows for additional combinations of features, which can differ significantly from the ones used to build the models in our study. In fact, it is likely that the combination used for the level of shell, Boolean operations, fillets, and drafts was not the most efficient one. In addition, we also tested different ways to combine the Boolean operations and the order of the fillets, which significantly affect the regeneration time. Therefore, the results for this part of the study may have been influenced by the assumptions considered during the modeling process.

It is also important to note that the results of our study are based on only two possible outcomes in the generation of geometric variations: successful and unsuccessful. However, there are various design scenarios in which a particular geometry might be generated with no errors but represent an incorrect and/or invalid model. These possible scenarios are described as follows:

- The geometric model generates successfully, and design intent is maintained. This is the desirable case.
- The geometric model regenerates successfully but design intent is not maintained. This case is especially problematic as some design intent errors may easily go unnoticed, as illustrated in Fig. 11. The left figure shows the initial state before modifications. The parameter to be varied in this case is the length of the connecting rod. When the geometry is changed by reducing the distance, the hole and the external cylindrical face of the part cease to be concentric (Fig.11. Right). This outcome is due to the disassociation that occurs between the sketches of the hole and the external shape of the upper axis. As a result, the parameter that controls the length of the connecting rod is duplicated to constrain the

sketches of the internal and external circles. For this reason, the solution space deviates from the original design intent.

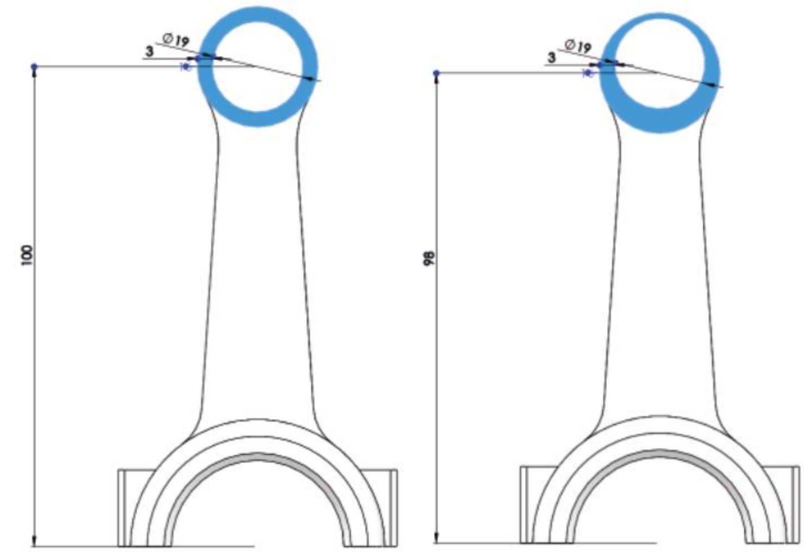


Fig. 11. Example of a model that regenerates correctly but does not maintain design intent (i.e., concentricity of the hole and the cylindrical face).

- The geometric model regenerates (partially) in a stable manner and design intent is not maintained. Some references (but not all) of a fillet/chamfer feature may be lost during modification or part of the feature cannot regenerate in a precise area. In our study, the software's feedback in this case is a warning. Therefore, it is counted as a successful scenario due to stable regeneration.
- The geometric model regenerates successfully but at a lower level of parametric quality. Design intent may or may not be maintained. Certain features may go missing or become redundant or even unnecessary. For example, Fig.13 illustrates a subtraction feature to create a cavity to accommodate the head of a bolt. When the distance between the bolt holes increases, as shown in Fig.12. right, the model maintains its design intent. However, the subtraction feature used to round up the vertical face to make room for the bolt heads becomes unnecessary, which causes the model to become not concise, according to the dimensions of CAD quality proposed by Company et al. [46].

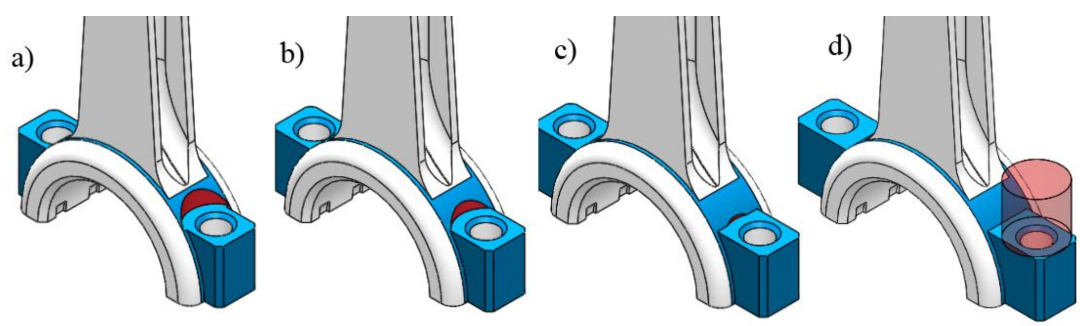


Fig. 12. The subtraction feature in different variations and in the "d" case without material to subtract.

6. Conclusions and future work

In this paper, we generated a set of parametric models of various parts according to different modeling methodologies to determine how the internal structure of the resulting models impacts the automatic generation of geometric variations and affects design intent. A deliberate effort was made to ensure all modeling strategies were applied rigorously and accurately, and the models were evaluated in terms of robustness and flexibility.

Our results showed that a less complex graph in a parametric model does not necessarily mean that the model is more robust, as observed with horizontal and resilient models. As the geometric complexity of a model increases, the number of features also tends to increase, making it exceedingly difficult to control and process the network of dependencies between features. In general, it was shown that the horizontal modeling methodology is not an efficient strategy for automated environments and the resilient methodology is most effective for simple parts without shelled features and scenarios that involve manual changes. The Explicit Reference Modeling methodology appears to be the most effective strategy to create robust models when the complexity is high, but it must be refined further so the intrinsic variability (i.e., the level of variability in the resulting parametric model due to the interpretation and execution of the methodology) is reduced and more detailed modeling cases are addressed. It is of interest to explore the level at which fillets and rounds need to be created in the design tree, when to apply them, how to handle shelled features, and how to determine the most robust Boolean scheme, are factors that have yet to be analyzed.

Although the results of our study are quantitative, our variable is dichotomous (successful/unsuccessful), which limits the level of detail of our observations. For more comprehensive measures of robustness and flexibility, it would be necessary to extract, analyze, and quantify all the errors in each modeling scenario and calculate the effort required to complete the design changes. The latter observation is of particular interest as we speculate that explicit and resilient strategies do not require the same level of effort to achieve the same degree of model robustness.

Finally, we found that formal modeling methodologies can still yield different results depending on how they are interpreted or because detailed explanations are not provided as to how to solve specific cases. Methodologies have an intrinsic variability due to not being concise. As future work, we are interested in performing similar studies with a more diverse set of models created by different users to determine the time required to reach different levels of robustness with each methodology. In addition, it would be interesting to conduct similar studies with different CAD software, especially with systems that have different underlying solid modeling kernels, to determine how different software handles the model regeneration process.

References

[1] Bodein Y, Rose B, Caillaud E. Explicit reference modeling methodology in parametric CAD system. Comput Ind 2014;65:136–47. https://doi.org/10.1016/j.compind.2013.08.004.

[2] Camba J, Contero M, Company P. Parametric CAD Modeling: An Analysis of Strategies for Design Reusability. Comput Des 2016;63:18–31. https://doi.org/10.1016/j.cad.2016.01.003.

[3] Cheng Z, Ma Y. A functional feature modeling method. Adv Eng Informatics 2017;33:1–

15. https://doi.org/10.1016/j.aei.2017.04.003.
[4] Camba JD, Contero M, Company P, Hartman N. The cost of change in parametric modeling: A roadmap. Comput Aided Des Appl 2021;18:634–43. https://doi.org/10.14733/cadaps.2021.634-643.
[5] Rynne A, Gaughran WF. Cognitive modelling strategies for optimum design intent in Parametric Modelling (PM). 2007.
[6] Amadori K, Tarkian M, Ölvander J, Krus P. Flexible and robust CAD models for design automation. Adv Eng Informatics 2012;26:180–95. https://doi.org/10.1016/j.aei.2012.01.004.
[7] Company P, Naya F, Contero M, Camba J. On the Role of Geometric Constraints to Support Design Intent Communication and Model Reusability. Comput Aided Des Appl 2020;17:61–76. https://doi.org/10.14733/cadaps.2020.61-76.
[8] González-Lluch C, Company P, Contero M, Pérez-López D, Camba JD. On the effects of the fix geometric constraint in 2D profiles on the reusability of parametric 3D CAD models. Int J Technol Des Educ 2019;29:821–41. https://doi.org/10.1007/s10798-018-9458-z.
[9] González-lluch C, Plumed R, Pérez-lópez D, Company P, Contero M, Camba JD. Computers in Industry A constraint redundancy elimination strategy to improve design reuse in parametric modeling 2021;129.
[10] Chase S, Murty P. Evaluating the Complexity of CAD Models as a Measure for Student Assessment. Eternity, Infin. Virtuality Archit. Proc. ACADIA 2000, 2000, p. 173–82.
[11] Summers JD, Shah JJ. Mechanical engineering design complexity metrics: Size, coupling, and solvability. J Mech Des Trans ASME 2010;132:0210041–02100411. https://doi.org/10.1115/1.4000759.
[12] Johnson MD, Valverde LM, Thomison WD. An investigation and evaluation of computer-aided design model complexity metrics. Comput Aided Des Appl 2017;15:61–75. https://doi.org/10.1080/16864360.2017.1353729.
[13] Bhaskara S. Analysis and visualization of complex computer aided design models as a design structure matrix. Invest Vis - Proc 13th Int DSM Conf 2011:61–75.
[14] González-lluch C, Company P, Contero M, Camba D, Plumed R, Camba JD, et al. A survey on 3D CAD model quality assurance and testing tools. CAD Comput Aided Des 2017;83:64–79. https://doi.org/10.1016/j.cad.2016.10.003.
[15] Mocko GM, Paasch R. Incorporating Uncertainty in Diagnostic Analysis of Mechanical Systems. J Mech Des 2005;127:315–25. https://doi.org/10.1115/1.1829071.
[16] Weber C. What is “complexity”? Procedings ICED 05, 15th Int. Conf. Eng. Des., Melbourne, Australia: 2005, p. 1–12.
[17] Hölttä KMM, Otto KN. Incorporating design effort complexity measures in product architectural design and assessment. Des Stud 2005;26:463–85. https://doi.org/10.1016/J.DESTUD.2004.10.001.
[18] Hartman NW. Defining expertise in the use of constraint-based CAD tools by examining practicing professionals. ASEE Annu Conf Proc 2005:2763–75. https://doi.org/10.18260/1-2--13970.
[19] Verein Deutscher Ingenieure. VDI 2209, 3D product modelling - Technical and organizational requirements Procedures, tools, and applications - Cost-effective practical use 2009:165.

[20] Johnson MD, Diwakaran RP. An educational exercise examining the role of model attributes on the creation and alteration of CAD models. Comput Educ 2011;57:1749–61. https://doi.org/10.1016/j.compedu.2011.03.018.
[21] Mathieson JL, Summers JD. Complexity metrics for directional node-link system representations: Theory and applications. Proc ASME Des Eng Tech Conf 2010;5:13–24. https://doi.org/10.1115/DETC2010-28561.
[22] Otto HE, Mandorli F. A Framework to Support 3D Explicit Modeling Education and Practice. Comput Aided Des Appl 2015;12:104–17. https://doi.org/10.1080/16864360.2014.949581.
[23] Kirstukas SJ. A Preliminary Scheme for Automated Grading and Instantaneous Feedback of 3D Solid Models. 68th EDGD Midyear Conf Proceeding 2013:53–8.
[24] Goldberg A V., Tarjan RE. A New Approach to the Maximum-Flow Problem. J ACM 1988;35:921–40. https://doi.org/10.1145/48014.61051.
[25] Watts DJ, Strogatz SH. Collective dynamics of 'small-world' networks. Nat 1998 3936684 1998;393:440–2. https://doi.org/10.1038/30918.
[26] Davis D. Quantitatively Analysing Parametric Models: Http://DxDoiOrg/101260/1478-0771123307 2014;12:307–19. https://doi.org/10.1260/1478-0771.12.3.307.
[27] Mowshowitz A, Dehmer M. Entropy and the Complexity of Graphs Revisited. Entropy 2012, Vol 14, Pages 559-570 2012;14:559–70. https://doi.org/10.3390/E14030559.
[28] Li W, Schuurmans D. Modular community detection in networks. IJCAI Int Jt Conf Artif Intell 2011:1366–71. https://doi.org/10.5591/978-1-57735-516-8/IJCAI11-231.
[29] Aranburu A, Justel D, Angulo I. Reusability and flexibility in parametric surface-based models : a review of modelling strategies. Comput Des Appl 2021;4:864–74. https://doi.org/10.14733/cadaps.2021.864-874.
[30] Landers DM, Khurana P. Horizontally-Structured CAD/CAM Modeling for Virtual Concurrent Product and Process Design (75) 2004.
[31] Gebhard R. 122 - A Resilient Modeling Strategy. Solid Edge University 2013 - Siemens; 2013.
[32] Bodein Y, Rose B, Caillaud E. A roadmap for parametric CAD efficiency in the automotive industry. CAD Comput Aided Des 2013;45:1198–214. https://doi.org/10.1016/j.cad.2013.05.006.
[33] Wang Y, Nnaji BO. Geometry-based semantic ID for persistent and interoperable reference in feature-based parametric modeling. Comput Des 2005;37:1081–93. https://doi.org/10.1016/j.cad.2004.11.009.
[34] RMS. Resilient Modeling Strategy: Solids Modeling Best Practices Manual. V9 2018. learnrms.com.
[35] Aranburu A, Justel D, Contero M, Camba JD. Geometric Variability in Parametric 3D Models : Implications for Engineering Design. Procedia CIRP 2022:383–8. https://doi.org/10.1016/j.procir.2022.05.266.
[36] Khan S, Gunpinar E, Moriguchi M, Suzuki H. Evolving a psycho-physical distance metric for generative design exploration of diverse shapes. J Mech Des Trans ASME 2019;141:1–13. https://doi.org/10.1115/1.4043678.
[37] Nerenst TB, Ebro M, Nielsen M, Eifler T, Nielsen KL. Exploring barriers for the use of FEA-based variation simulation in industrial development practice. Des Sci 2021;7:1–22. https://doi.org/10.1017/dsj.2021.21.

[38] Jackson C, Prawel D. The 2013 State of 3D Collaboration and Interoperability Report. 2013.
[39] Mountstephens J, Teo J. Progress and challenges in generative product design: A review of systems. Computers 2020;9:1–23. https://doi.org/10.3390/computers9040080.
[40] Khan S, Awan MJ. A generative design technique for exploring shape variations. Adv Eng Informatics 2018;38:712–24. https://doi.org/10.1016/j.aei.2018.10.005.
[41] Krish S. A practical generative design method. CAD Comput Aided Des 2011;43:88–100. https://doi.org/10.1016/j.cad.2010.09.009.
[42] Khan S, Gunpinar E, Mert Dogan K. A novel design framework for generation and parametric modification of yacht hull surfaces. Ocean Eng 2017;136:243–59. https://doi.org/10.1016/j.oceaneng.2017.03.013.
[43] Wang R, Zhang J, Bian S, You L. A survey of parametric modelling methods for designing the head of a high-speed train. Proc Inst Mech Eng Part F J Rail Rapid Transit 2018;232:1965–83. https://doi.org/10.1177/0954409718756558.
[44] Otey J, Company P, Contero M, Camba JD. Revisiting the design intent concept in the context of mechanical CAD education. Comput Aided Des Appl 2018;15:47–60. https://doi.org/10.1080/16864360.2017.1353733.
[45] Suchindran C. Sample Size. In: Elsevier, editor. Encycl. Soc. Meas., 2005, p. 437–41. https://doi.org/10.1016/B0-12-369398-5/00057-8.
[46] Company P, Contero M, Otey J, Plumed R. Approach for developing coordinated rubrics to convey quality criteria in MCAD training. CAD Comput Aided Des 2015;63:101–17. https://doi.org/10.1016/j.cad.2014.10.001.