International Journal of Industrial Engineering and Management (IJIEM), Vol. 9 No 4, 2018, pp. 177-186
Available online at www.iim.ftn.uns.ac.rs/ijiem_journal.php
ISSN 2217-2661

http://doi.org/10.24867/IJIEM-2018-4-177

Solution Space Development: Conceptual Reflections and Development of the Parameter Space Matrix as Planning Tool for Geometry-based Solution Spaces

Paul Christoph Gembarski
Leibniz University Hannover (Senior Researcher, Institute of Product Development, Welfengarten 1A, 30167 Hannover, Germany, gembarski@ipeg.uni-hannover.de)

Roland Lachmayer
Leibniz University Hannover (Professor, Institute of Product Development, Welfengarten 1A, 30167 Hannover, Germany, lachmayer@ipeg.uni-hannover.de)

Received (18.10.2018.); Revised (21.11.2018.); Accepted (03.12.2018.)

Abstract

Today's CAD-systems offer the possibility to model geometry-based solution spaces based on parametrics and feature technology. Here, the solution space is the set of all feasible product alternatives from which a distinct variant for a defined set of requirements may be configured. A necessary step prior to modelling the solution space is to acquire knowledge about dependencies of requirements, solutions and restrictions that are dictated by the supply chain, e.g. manufacturing restrictions. In this article, the authors contribute to this field by developing the Parameter Space Matrix (ParSM) as a tool for a structured elicitation of requirements, solution space restrictions and the resulting model parameters for the CAD-model. Furthermore, the application of ParSM is shown and discussed on a toaster with variable body elements where the manufacturing restrictions result of an additive manufacturing process.

Key words: *Parameter Planning, Parameter Space Matrix, Solution Space Development, Product Configuration*

1. INTRODUCTION

The steady increase of competitiveness, innovation and complexity in many companies has driven the progress of computer-aided design (CAD) and computer-aided engineering (CAE) tools in the recent 20 years. Of particular interest for adaptive and variant design is the use of parametric design systems, in which not only the shape of a component or an assembly is modelled, but also their describing parameters [1]. Moreover, constraining parameters mathematically or logically allows implementing explicit knowledge like dimensioning formulae or design rules within the digital prototype [2]. So, in relation to adaptive and variant design, the designer does not only specify the product shape of a single component but also the control and configuration concept and thus describes a solution space [3, 4].

1.1 **Motivation and Aim**

However, modelling parts and assemblies as solution space and not as distinct variants today is not the state of the art in many design departments. The need for a new variant or changes to the product lead to ever new CAD and CAE models. A reason for this is the effort required to plan parameters, their dependencies and the corresponding model structure. The more complex the geometry and the larger the assemblies, the more important it is to constrain model parameters and reference individual features to build robust CAD models [5-7].

Considering a mass customization business model, the design of the solution space is a key principle [8, 9]. The ability to cope with the resulting complexity is enabled by knowledge-based engineering (KBE) systems in general and product configurators in particular [4, 10-11]. Process models for creating KBE-applications and prior to that acquiring the necessary knowledge to be implemented are available (refer e.g. to [12, 13]). But although contemporary CAD-systems offer the possibility to implement knowledge in the digital prototypes themselves (e.g. refer to [2]), there is a lack of concrete modelling principles or detailed application examples. In this article, after conceptual reflections on solution space development, the authors narrow this gap by developing the Parameter Space Matrix (ParSM). ParSM is a tool for the structured elicitation of requirements, solution space restrictions and the resulting model parameters for the CAD-model in geometry based solution spaces.

1.2 **Structure of the Paper**

In the following Section 2, the theoretical background of design solution spaces and solution space development is reflected from literature. This is supplemented by a review of geometry-based solution space modelling in contemporary CAD-systems. In Section 3, setup and functioning of ParSM is described. Afterwards ParSM is applied in the case study "customizable toaster". Subsequently in section 4, the article is summarized, the implications of the case study on parametric CAD design in general is discussed and further research potentials are outlined.

2. THEORETICAL BACKGROUND

2.1 Design solution spaces

In order to get a fundamental understanding of solution spaces we analysed literature. The term “solution space” (and its German translation “Lösungsraum”) is used cross-disciplinarily so the literature review was narrowed on specific research fields that are considered to be relevant. To these belong design engineering, design methodology, design theory, product management and product configuration. Additionally, the mass customization literature was reviewed. As a result of the literature review, four different views on solution spaces and solution space development were consolidated that are presented in the following sub-sections. In order to be included in the review, the literature had to present detailed concepts or definitions on this topic. Not included was literature that just mentions solution space development as core competence in Mass Customization or as ability to define options from which a customer can choose his variant, without going further into detail.

2.1.1 *External Variety View*

One of the first appearances of the term “solution space” is within the works of Hubka on the theory of machine systems [14]. There, he understands the set of machine systems that perform a given set of functions as solution space. Ponn expands the term in relation to a given design task and introduces the requirement space as a set of all development goals and required product characteristics [15]. In the product development process, the required product properties are compared with the properties of the designed artefact and approximated by synthesis analysis loops (Fig. 1).

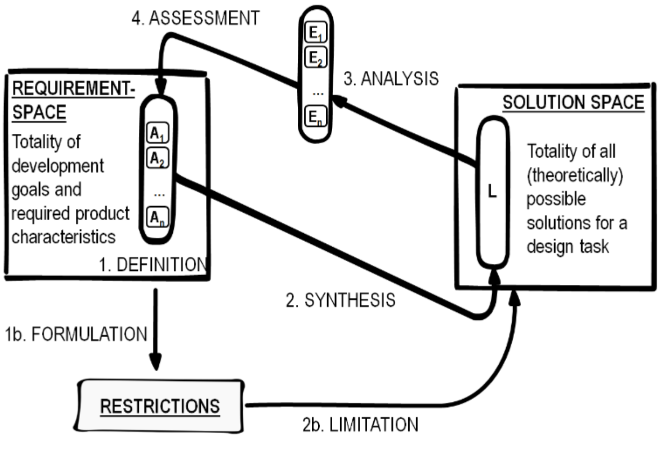

Figure 1. Requirement space and solution space in the development process (adapted acc. to [15])

Since not all areas of the solution space are accessible due to various restrictions, e.g. from manufacturing, the formulation of such restrictions is an accompanying concept for setting up the solution space [16]. This will be presented in more detail in sub-section 2.1.5.

Different approaches like the Munich Concretization Model [17] apply a further structure on the solution space that corresponds to standard development processes like the German VDI2221 [18] or the mechanical design process according to Ullman [19]. In the mentioned model, the solution space is divided into a functional level that is used to decompose functions into sub-functions and their relations. The effect level then links concepts and solution principles. Finally, on the design level, the assembly structure is specified.

In the end, this **external variety view** leads to a set of feasible product variants that fulfil a certain set of (customer) requirements [20]. Depending on the approach, interactions and relations between solution space and requirement space are discussed but only on conceptual levels or with reference to requirement engineering. Planning external variety incorporates identifying and clustering of customer needs and strategic market planning and is not further considered in this article.

2.1.2 *Internal Variety View*

The above-mentioned relationships between requirements and solutions are integral part of the design method Axiomatic Design [21]. This approach is based on a domain concept: The customer domain has to be understood as the set of all customer requirements, while the functional domain contains functional requirements, which already represent a solution-neutral translation of customer needs into the language of the designer. The third domain, the physical domain, covers design parameters as representation of a design solution that is suitable for a functional requirement. These also represent the components of a system at the highest hierarchical level. These can, for example, be further decomposed into distinct dimensions of effective areas, measurements, etc. Content of the final fourth domain are process variables that characterize the core parameters of the manufacturing processes, with which a design parameter is realized (Fig. 2).

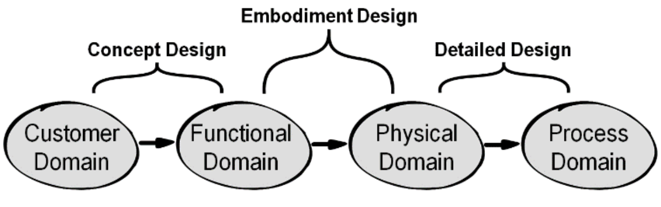

Figure 2. Domain Concept of Axiomatic Design (acc. to [21])

The development process in Axiomatic Design is strongly structured and formalized by the domains. The basic principle here is that the requirements of a predecessor domain are mapped to the solutions of the following domain using design matrices (shown in Fig. 3 exemplified for functional requirements and design parameters of a skip loader).

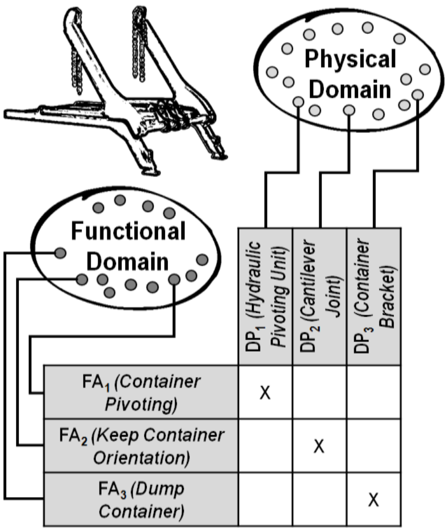

Figure 3. Mapping of functional requirements to design parameters in a design matrix

An important principle here is that the design problem is gradually decomposed into sub-problems. In the domain model, this leads to iterative zig-zagging between two adjacent domains until exactly one solution can be assigned to a single requirement. For the example of the skip loader, this means that the hydraulic pivoting unit is decomposed down to the individual parameters such as size of the ring surface, stroke of the cylinder, etc.
There are two axioms to be considered that gave the method its name: First, the independence axiom implies that in an ideal design after decomposition, a design parameter can only be assigned to a single functional requirement. In this way, it is ensured that functional requirements are not mutually exclusive and that no cyclical dependencies arise.
If alternative design matrices exist, the information axiom states that the one design with the lowest information content should be favoured. The information content's calculation is grounded on Shannon's information-based entropy and is considered a measure of structural complexity in information technology [22]. The approach of axiomatic design is also discussed in context of product family design, e.g. in the work of Jiao et al. [23].
Another domain concept as description for solution spaces is proposed by Aldanondo and Vareilles [24]. There, instead of requirements, selectable product features and their characteristics are formulated and matched to product components or features in the design domain (Fig. 4). The third domain contains the manufacturing processes used to produce and assemble the individual product variants. In turn, a process chain can be assigned resources such as production equipment and processing time. Aldanondo and Vareilles explicitly focus on the product portfolio by providing a solution space with a description of all available end product variants based on components. Properties, components and process chains are formulated as a constraint network. The goal of Aldanondo and Vareilles is joint product and production process configuration.

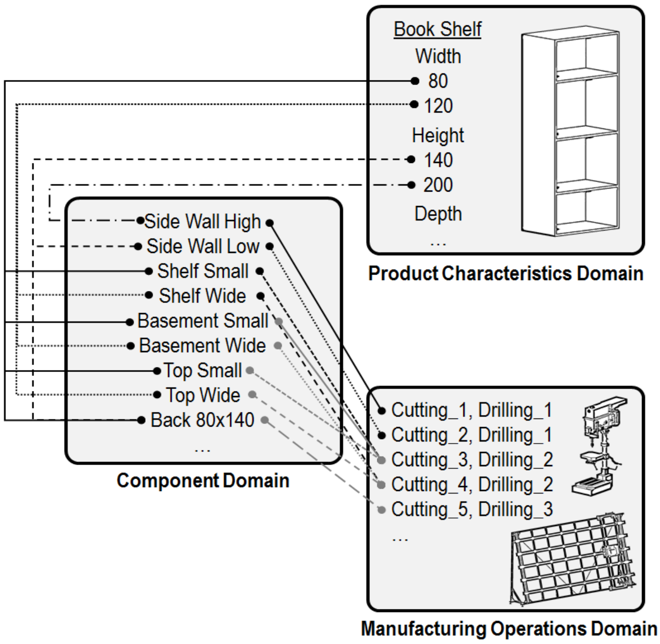

Figure 4. Combined Product and Process Configuration acc. to [24]

Approaches that cover this **internal variety view** on a solution space usually include models for the relationships between solution space, requirement space and the value chain configuration space that later has to realize the individual product variants.
There exists a number of planning aids for the internal variety that usually take into account product architecture and order penetration point. As an example stand the modularization method modular function deployment that was presented by Ericcson and Erixon [25] or the variant tree invented by Schuh [26].

2.1.3 *Exploration View*

Basically, authors that describe approaches for solution space exploration use a mix of external and internal variety view. For example, Lenders [27] and Lüdtke [28] base their approaches on set-based concurrent engineering like used at Toyota. The basic idea of this design methodology, which until now has been used only sporadically in industry, is not to make an early determination of a single solution concept, but to consciously pursue several parallel concepts and to define requirement value ranges instead of requirements with fixed values. At the evaluation gates, only those concepts are excluded in which a certainty of the requirement violation can be predicted, all other concepts are to be further detailed and then re-evaluated at the next decision gate.
So, in the understanding of the exploration view the solution space is a search space for an artefact to be developed. The solution space converges as the development process progresses. As a consequence, most approaches that cover this specific view provide tools for concept evaluation or decision making. From a KBE point of view, this contributes to reasoning mechanisms for design automation in the concept phase.

2.1.4 *Degree-of-Freedom View*

The above concepts understand the solution space as a set which contains a design that best meets the given requirements. Gero [29] reverses this: Following his argumentation, a design prototype represents a space where a design artefact, regardless whether product, subassembly or single part, may be altered in a certain way (Fig. 5).

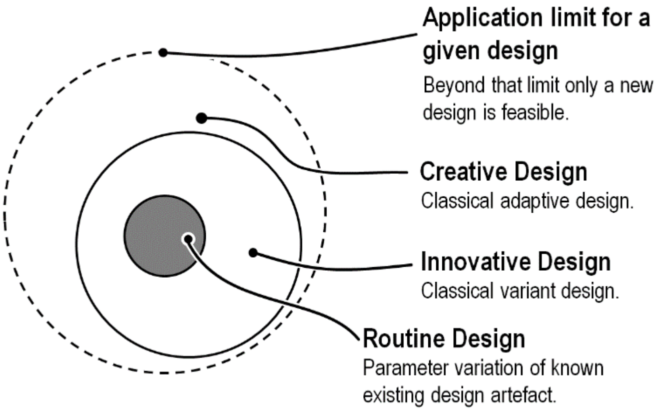

Figure 5. Design Prototypes (acc. to [29])

One way to do this is changing a product's parameters and then to regenerate the design which is introduced as routine design. In contrast to that, innovative and creative designs represent the traditional approaches to variant and adaptive design. The limit of creative design also marks the end of the variation possibilities of a given design. Beyond that limit only a new design may satisfy the requirements. Years before parametric CAD-systems became standard in the design departments Gero had been postulating renowned principles of computer aided design, namely parametrics, feature-based design and templates [1].
Defining degrees-of-freedom regarding the product shape or regarding the occurrence of product components is discussed also with regard to planning of variance. Two examples are the creation of product templates according to Cox [30] and forward variance planning according to Gembarski et al. [31]. Other degrees-of-freedom may result from feature occurrences, materials or surface finishes [32].

2.1.5 *Restricting a Solution Space*

Usually, not all areas of the solution space are accessible during the development process because multiple restrictions have to be considered. Restrictions originate from various circumstances and ensure that a product variant which follows all of them is valid [16].
There are four major groups of restrictions. The first, geometric restrictions refer to the topology, shape, dimension, number or sequence of features or components. Specifically, design interfaces, the available installation space, preferred standards or cross-sections of semi-finished products can be limited in this way [33].
Second, functional restrictions affect the interaction of the individual components within the product. Among other things, this ensures that the material, energy and information flows between two adjacent components are compatible with each other or that certain solution principles are excluded from the beginning of the design process (e.g. no use of oil hydraulics or combustion engines) [34].
Third, technological restrictions result from product structuring and availability of manufacturing technologies. Related to an individual manufacturing facility, travelling distances of a milling machine or the maximum hardening depth of an induction hardening process define such limits of the solution space [1]. Moreover, handling processes and logistics as well as disassembly and recycling dictate restrictions like maximum weight of a component, clearance and joining technology to name only a few [16]. In product development this is considered in the so called design for X-approaches [e.g. 35-36].
The fourth and final group of restrictions is economic ones, e.g. available resources in product development and use of existing components of predecessor projects [33].

2.2 **Solution space Modelling in CAD-Systems**

The basis for modelling a geometry-based solution space in a CAD-system for mechanical design is its ability to differentiate between shape and its describing parameters [3].

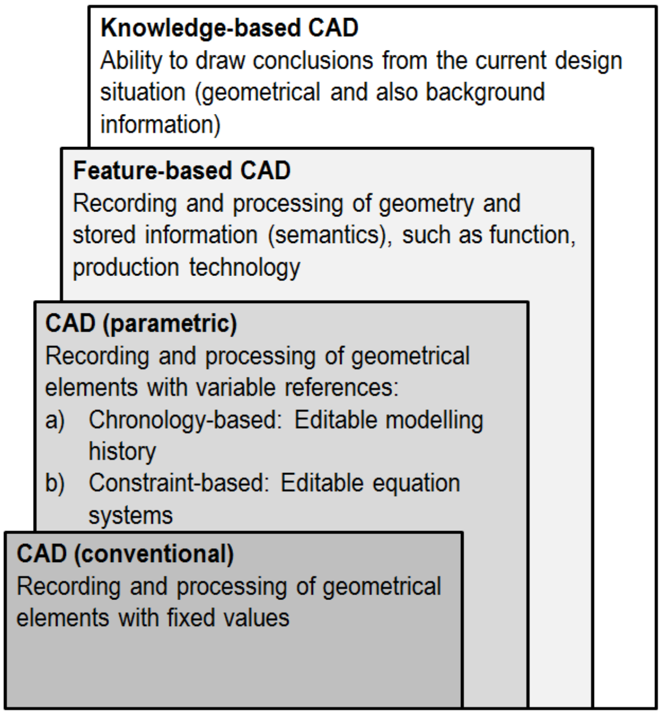

Figure 6. Overview of the principles of 3D modelling [37]

In parametric CAD, parameters can be related by mathematical and logical constraints that establish editable equation systems [2]. Furthermore, chronology-based references determine the genesis of the model and thus the sequence of all the individual operations for geometry creation and modification [37]. By defining such dependencies and user-defined parameters, it is possible to explicitly implement design knowledge in a CAD model [38].

The German VDI guideline 2209 [37] mentions two other types of CAD systems, which provide additional functionality for creating variable geometry models and for mapping design knowledge, which are feature-based and knowledge-based CAD systems (Fig. 6). Feature-based ones are an extension of parametric systems. In this context, a feature represents a semantic information object that is usually formed from several contiguous geometry elements with parametrics and behavioural rules [6]. As a result, features can adapt themselves to their environment to a limited extent. Knowledge-based engineering (KBE) goes a step further in order to adapt a designed artefact even more easily to new functional or design requirements. Hirz emphasizes that 'knowledge-based design supports design processes by reusing predefined methods, algorithms or results, and it is integrated into specific tasks or workflows that are involved in the design processes' [6].

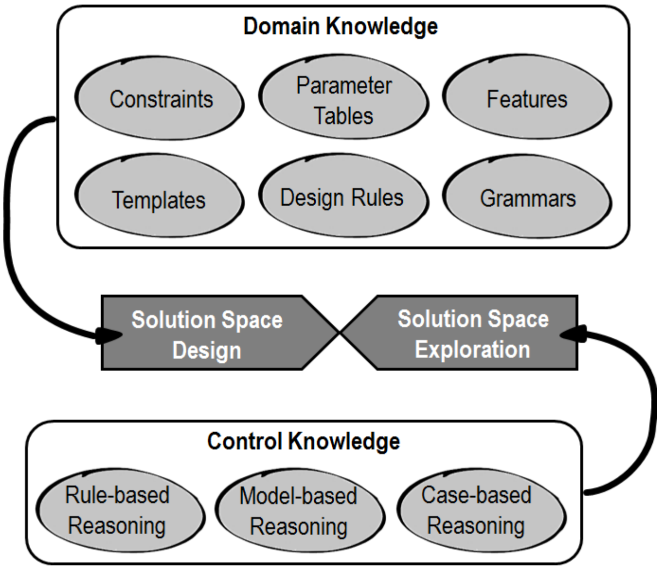

Figure 7. Knowledge Modelling in KBE and KBD

In detail, two different kinds of knowledge have to be considered (Fig. 7): First, domain knowledge describes a solution space in which a solution for a design problem may be found [38]. This domain knowledge may be formulated e.g. by constraints between parameters that express dimensioning formulae [1]. Parameter tables represent part families where geometry is instantiated with the dimensions and other parameters in one row of the table [2]. The feature-concept was already mentioned, templated aggregate multiple features and rigid geometry in a reusable, updatable building blocks in a virtual prototype [30]. Especially geometric templates may be considered as design prototypes for routine or innovative design activities in the sense of Gero. These provide variable geometry and configuration parameters [6]. Such a template can be used as starting point for detailed design respectively. Design rules are if-then-else-statements that are fired procedurally. They can be used e.g. to control the occurrence of product components in relation to parameter values. Grammars may be understood as graph language and also use rules but in a different way. Given is a representation of geometry, either as an abstract graph or as real 2D or 3D geometry [40]. Applying the rules by the synthesis algorithms now means searching for defined structures in the geometry representation and replacing them by structures that are described in the rule. In this way, a huge number of design alternatives may be set up and explored very rapidly [41].

Second, control knowledge determines the way a solution space is explored [42]. There are three major types of inference engines that contain the control knowledge:

- Rule-based: The knowledge representation is based solely on if-then-else statements. These rules may be related to each other, e.g. for initiating subordinate rules or eliminating other rules from the working memory. Many authors report that purely rule-based systems are only suitable for use with local and narrowly defined problems. This is due to the fact that with an increasing number of artefacts and rules, these systems suffer bad maintainability (e.g. refer to [43, 44]).
- Model-based: The limitation of the design solution space is based on a product model consisting of the system components and their relationships [10]. The relationships may be e.g. physically or logically (constraint-based) or on the basis of resource allocation and resource consumption [45].
- Case-based: Here, no explicit configuration rules or models are needed. The reasoning is made on the basis of previously recorded solutions as cases (refer e.g. to [46]). Depending on the maturity of the inference mechanism, the system can either only find solutions that exactly match to a given requirement profile, or make a selection of several cases representing the best-fit. Highly developed systems are able to alter or combine existing cases in order to derive new solutions [47].

3. THE PARAMETER SPACE MATRIX (ParSM)

For an initial development of a solution space, where variable product models have to be established that use techniques of KBE, two different situations have to be considered: (1) a predecessor product is available and can be used as basis for solution space development; (2) the product development process starts right at the beginning with no preconditions but the customer and / or functional requirements.

In the first case, the degree-of-freedom view of the solution space is a possible starting point. Since the shape of the product is already modelled, most of the CAD-model's parameters are already known and determined. Then, the design task is to unfix them according to existing restrictions and to develop an accurate configuration sequence. In the second case, the internal variety view, which leads to the laws-of-creation of a design, is the starting point. Usually, geometric models are used already at a concept or draft level. Here, the challenge is to decompose this concept

REQUIREMENTS					MODEL PARAMETERS OF COMPONENT: ***BASE PLATE 1220.776.A***				RESTRICTIONS				
1.1	1.2	2.1	2.2	...					A1	A2	M1	M2	...
Withstand Force of ... N	Maximum Deformation: < ... mm	In-House Manufacturing	Use available jigs and clamping tools	...					Assembly: Connector 1272.310.A	Assembly: Connector 1254.200.C	Travelling distance CNC Milling Machine 4022	Dimensions of Clamping Turret 9012.221.A	...
1	1	1	1		NAME	UNIT	VALUE	COMMENT			2.1	2.2	
2500					P:Force	N	2500	Applied Force					
	4				P:f	mm	4	Maximum Deformation					
		x	x		P:L	mm	600	Bounding Box Length			< 800	< 720	
					P:L_AB	mm	220	Length AB to Connector	< 225	< 250			
		x	x		P:D	mm	182	Bounding Box Depth			< 600	< 320	
					P:S	uL	2	Number of Stiffening Elements					
					P:X_Dr1	uL	TRUE	Occurence of Hole Pattern1	FALSE	TRUE			
					...	...	...	...					

Figure 8. Parameter Space Matrix

geometry to the later design parameters like introduced in axiomatic design without violating any constraints. So, the design task is to determine the relationships of requirements, model parameters and restrictions.

3.1 **Developments of ParSM**

Since until now no computer-aided tools for the joint elicitation of requirements, parameter hierarchies and restrictions exist, we developed the Parameter Space Matrix (ParSM). As requirements for its development, both of the above situations should be supported by ParSM, a direct implementation into a CAD-environment should be possible, an integration of spreadsheet functions should be used and mechanisms for conflict resolutions should be available.

Basically, ParSM is an extended parameter table of the later CAD-model (Fig. 8). Depending on the CAD-system, ParSM can be created within the CAD environment or in a spreadsheet application. The following description bases on an Excel macro spreadsheet. The headlines of sub-sections 3.1.1 to 3.1.3 correspond to the main column headlines in the figure.

3.1.1 *Centre Part: Model Parameters*

The central element of the matrix is the parameter list of the component (Fig. 8). Here, the model parameters of the part (i.e. dimensions and feature parameters) are recorded. The notation of the parameters in the illustrated table is according to Autodesk Inventor in which the examples have been modelled. Here, a parameter is described by name, unit of measure, value and comment.

Depending on the CAD-system, different parameter types may be available. In Inventor, numeric, text and Boolean parameters are present. Especially the latter are important to control e.g. the occurrence of a feature that determines if the feature is active or suppressed. As unit of measure, Inventor offers basically all physical units with all suitable prefixes. This includes units for length (mm, inch, nautical mile, etc.), angularity (radian, degree) but also for mass, forces, power, velocities, electrical or luminosity.

Values are user inputs or calculated by the matrix based on mathematical constraints. Usually, in CAD-systems only a limited count of mathematical operators is available. The use of a spreadsheet application extends this.

*=Buildspace X - 2*Wall Thickness - Installation Space - 80*

NAME	UNIT		
...	...	...	...
P:TS	mm	6	Wall Thickness
S:L1	mm	280	Buildspace X
S:L2	mm	60	Installation Space
P:L_s	mm	128	Bounding Box Depth
...	...	...	...

Figure 9. Parameter Constraining

Such a constraint is shown in Fig. 9 where P:L_s is calculated based on other parameter values from the ParSM. In order to distinguish user input and derived parameters it is beneficial to use different fonts. Furthermore, prefixes in the parameter names may help organizing parameter hierarchies (like the "P:" that indicates a parameter that effects a part; the “S:” effects the whole assembly / system).

3.1.2 *Left Wing: Requirement Specification*

To the left of the parameters are the requirements and their influence on the model parameters (Fig. 8). To have a better organization, requirements should be numbered the same way as done in the specification list. If the applications allow this, both documents can be linked, so that the requirements are passed and updated automatically in ParSM. Additionally, requirements may be prioritized which is beneficial for later conflict resolution. In the example above, all requirements have priority 1. In order to document the relationships between requirement and parameter, assignments, numeric values or relevant formulas may be entered in the crossing fields. As can be seen, applied force and maximum deformation (1.1 and 1.2) both are directly linked to the model parameters P:Force and P:f, 2.1 and 2.2 take an influence on P:L and P:D.

3.1.3 *Right Wing: Restriction List*

The right part of ParSM is dedicated to the restrictions (Fig. 8). Also the restrictions should be indexed for better organization (A1 and A2 are restrictions that result from other components that have to be assembled to the ground plate; M1 and M2 are manufacturing restrictions). Additionally, an assignment to a restriction is possible.
As can be seen from the example in Fig. 8, the travelling distances of a specific milling machine (M1) and the width / height of a clamping turret for multiple part processing (M2) restrict the maximum dimensions of the baseplate. Both restrictions result from the requirements for in-house manufacturing and the favourable use of existing jigs and clamping tools.

3.1.4 *Conflict Detection and Resolution*

Bookmarking requirements, model parameters and restrictions in one common matrix allows also detecting and resolving conflicts. The simplest conflict that can occur is the violation of a restriction. For its detection, ParSM is equipped with a macro that checks conformity with all restrictions in the same row with a parameter. The macro is fired every time when a parameter value is entered or ParSM is saved and informs the user about the violation. As restrictions can be seen as value range limits, the information dialog also gives out the allowed borders.
A conflict of requirements and the resulting conflicting restrictions can be resolved when the priorities are adapted. In the example in Fig. 8, the requirements 2.1 and 2.2 both restrict the value range of different parameters. If P:L should be 770 mm this would violate 2.2, but the travelling distances of the milling machine would allow this measurement. The user can downgrade requirement 2.2 to priority 2, which would mean that new or modified jigs would be acceptable now. ParSM then does not take this violation into account.

3.2 **Application Example: Modular and Adaptable Toaster Housing**

This sub-section presents the application of ParSM for the planning of an adaptable toaster housing (Fig. 10). A modular design was chosen, so that either design department or end customers are able to configure new toaster designs. Additionally, upper and lower housing part as well as bottom and top covers have multiple degrees-of-freedom regarding the shape. Up to 6 slots can be selected and modified in dimensions and orientation. The cross-sections of the housing parts can be adapted in dimensions and rounding which affects the curvature of the entire housing. In case of an individualized toaster, the housing parts will be manufactured in ABS plastics on a laser sintering machine. Process restrictions, e.g. minimal wall thicknesses or the dimensions of the process chamber, have to be considered in ParSM. Additional to the shape, the colour can be chosen from a given list since the processed parts are dip-coated. Some example configurations are depicted in Fig. 11.

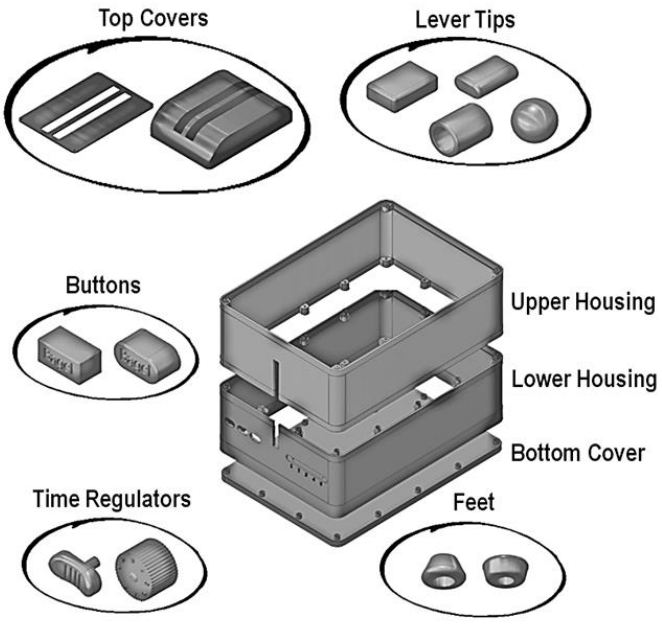

Figure 10. Modular adaptable Toaster Housing

Fig. 12 shows an excerpt of the ParSM established for the upper housing part. From an assembly point-of-view, the connecting interfaces to top cover and lower housing have to be considered, as well as the lever.

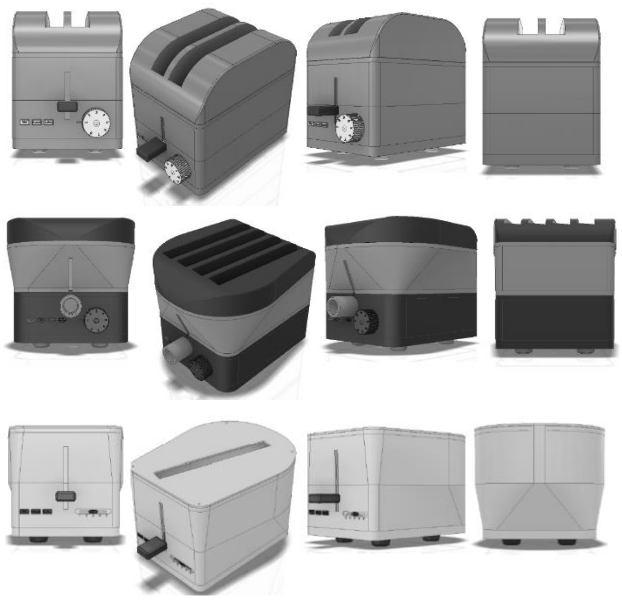

Figure 11. Housing Variations

REQUIREMENTS					MODEL PARAMETERS OF COMPONENT: *Upper Housing Part Toaster*				RESTRICTIONS					
1.1	1.2	2.1	2.2	...					A1	A2		M1	M2	...
Number of Slots	Max. Surface Temperature	Adaptable shape	Symmetry	...					Assembly: Housing Height	Assembly: Lever	...	Laser sintering - Process Chamber	Laser sintering - Downskin Angle	...
4	60°C	1	1		NAME	UNIT	VALUE	COMMENT	A1	A2		2.1	2.2	
		x			P:e1_x1	mm	-180	Upper Plane Edge 1				540		
x		x			P:e1_y1	mm	-120	Upper Plane Edge 1				480		
		x	x		P:e1_x2	mm	180	Upper Plane Edge 2				540		
x		x			P:e1_y2	mm	-120	Upper Plane Edge 2				480		
		x	x		P:e1_x3	mm	-180	Upper Plane Edge 3				540		
x		x			P:e1_y3	mm	120	Upper Plane Edge 3				480		
		x			P:e1_x4	mm	180	Upper Plane Edge 4				540		
x		x			P:e1_y4	mm	120	Upper Plane Edge 4				480		
		x			P:e1_H	mm	60	Upper Plane Height	60-80			280		
		x			P:e2_x1	mm	-180	Lower Plane Edge 1					18°	
x		x			P:e2_y1	mm	-120	Lower Plane Edge 1					18°	
					...	...	...	...						
		x			P:e1_r1	mm	20	Upper Plane Radius Left						
		x			P:e1_r2	mm	40	Upper Plane Radius Right						
					...	...	...	...						
					S:C_L	mm	8	Cutout Lever Width		6-10				
					S:C_H	mm	36	Cutout Lever Height		20-40				
					S:C_t	mm	0	Position from middle		-40; 40				
					...	...	...	...						

Figure 12. Parameter Space Matrix for Upper Housing Part (excerpt)

For variants in which the lower cross-section is smaller than the upper one, then a lever with a spring element has to be chosen since the tip of the lever must move out. With respect to manufacturing, the laser sintering technique does not imply a lot of relevant restrictions, since the powder bed is an accurate support structure for the built part. Moreover, the part is big enough so that minimum radii, minimum wall thicknesses etc. do not affect the geometry. Nevertheless, the process chamber of the production machine is a strong restriction since it limits the dimensions of the housing part.

4. DISCUSSION AND CONCLUSIONS

The present article discusses the state of the art of solution space development. In the first part, the article discusses the state of the art of solution space development. It does so by going through four different views on design of solution spaces, namely (1) External Variety View, (2) Internal Variety View, (3) Exploration View and (4) Degree-of-Freedom View.

In the second part, the article presents the developed parameter space matrix (ParSM) as our main contribution, a planning aid for parameter planning and constraining in CAD models.

Ideally, the parameters out of ParSM can be transferred directly as a set, e.g. through an Excel coupling, into the component to be designed [6, 38]. In principle, this creates an increased memory requirement of the individual components, because in addition to the model parameters, the imported ones must also be managed. In the existing implementations, however, this effect and the increase in rebuild time of the CAD model are negligibly small.

At first glance, modelling with ParSM seems to be more complicated than simplified because requirements and restrictions have to be modelled as well. However, this is only apparently the case because the model planning aspect is typically present in parametric CAD design of multi-variant products, but is rarely documented [3, 5]. An example of this is the various skeletal techniques used for the respective CAD systems [2]. ParSM can be a useful supplement here because it transparently describes the individual dependencies between the parameters and, by relating them to the respective requirements and restrictions, offers a (semi-automatic) decision support when conflicts arise during modelling, e.g. when value ranges are incompatible.

In simple contexts such as the geometry of the housing parts, the restriction check in ParSM can be performed independently of the CAD model. If KBE functions are offered by the CAD system, further options are available for a restriction check [2, 10, 16]. For example, the violation of a physical design space model can be checked by a collision analysis and the determination of the rigidity and modal properties of structural components could be inspected in a linked FEM system. In the case of a restriction violation, a reasoning mechanism (e.g., implemented design rules or a case base) may cause the model to change.

There are indeed further research potentials. One point is additional computer support in generating the restriction list. Joining parameters and requirements based upon the requirement specification is easy to implement, regarding the restrictions a library could be established. Then, if the user wants to add restrictions for a certain production technology or machine, he could just select the desired process from the library and include it in ParSM. Nevertheless, the manual assignment of parameters, requirements and restrictions is cumbersome and error-prone with increasing number of parts. So it has to be examined if also complex assemblies can be decomposed with acceptable effort. Since model set up and chronology of features is, application of machine learning or case-based reasoning might be limited but as starting point nevertheless helpful.

In addition, this approach implies that there is already a concrete idea of the product shape and its functioning, which is true with respect to the inner variety view and the degree-of-freedom view of the solution space [47]. Another interesting issue results from the exploration view: As it was seen in the literature review, most authors that research solution space exploration develop tools for early concept evaluation and assessment [27-28]. At least the restriction list of ParSM could be beneficial in this field too, since it clearly describes areas of the solution space that are not accessible in the development process.

Furthermore, regarding organizational constraints for solution space development, it is assumed, that the developer has rich knowledge not only about the product he is designing, but also about many technological aspects and in-depth manufacturing processes. Embedding this in the organisational design [e.g., 48] or the design process for mass customization could leverage additional potentials.

5. REFERENCES

[1] Vajna, S. (2009), *CAx für Ingenieure: eine praxisbezogene Einführung*, Springer, Berlin, Heidelberg, Germany.

[2] Gembarski, P.C., Li, H. and Lachmayer, R. (2015), "*KBE-Modeling Techniques in Standard CAD-Systems: Case Study – Autodesk Inventor Professional*", Proceedings of the 8th World Conference on Mass Customization, Personalization, and Co-Creation (MCPC 2015), Montreal, Canada, October 20th-22nd 2015, pp. 265 - 283.

[3] Shah, J.J. (2001), "*Designing with parametric cad: Classification and comparison of construction techniques*", Geometric Modelling – Proceedings of the Sixth International Workshop on Geometric Modelling, Tokyo, Japan, December 7–9, pp. 53 - 68.

[4] Gembarski, P.C. and Lachmayer, R. (2017), "*Designing Customer Co-Creation: Business Models and Co-Design Activities*", International Journal of Industrial Engineering and Management (IJIEM), Vol. 8, No. 3, pp.121 - 130.

[5] Weber, C. (2011), "*Design Theory and Methodology-Contributions to the Computer Support of Product Development/Design Processes*", The Future of Design Methodology. Springer, Berlin, Heidelberg, Germany, pp. 91 – 104.

[6] Hirz, M. (2012), *Integrated Computer-aided design in automotive development*, Springer, Berlin, Heidelberg, Germany.

[7] Gembarski, P.C., Li, H. and Lachmayer, R. (2017), "*Template-Based Modelling of Structural Components*", International Journal of Mechanical Engineering and Robotics Research, Vol. 6, No. 5, pp. 336-342.

[8] Salvador, F. De Holan, P.M. and Piller, F. (2009), "*Cracking the code of mass customization*", MIT Sloan Management Review, Vol. 50, No. 3, pp. 71-78.

[9] Suzić, N., Forza, C., Trentin, A. and Anišić, Z. (2018), "*Implementation guidelines for mass customization: current characteristics and suggestions for improvement*", Production Planning & Control, Vol. 29, No. 10, pp. 856–871.

[10] Hvam, L., Mortensen, N.H. and Riis, J. (2008), *Product Curtomization*, Springer, Berlin, Heidelberg, Germany.

[11] Reichwald, R. and Piller, F. (2009), *Interaktive Wertschöpfung*, Gabler, Wiesbaden, Germany.

[12] Schreiber, G., Wielinga, B., de Hoog, R., Akkermans, H. and Van de Velde, W. (1994), "*CommonKADS: A comprehensive methodology for KBS development*", IEEE expert, Vol. 9, No. 6, pp. 28-37.

[13] Stokes, M. (2001), *Managing engineering knowledge: MOKA: methodology for knowledge based engineering applications*, Professional Engineering Publishing, London, Great Britain.

[14] Hubka, V. (1984), *Theorie Technischer Systeme: Grundlagen einer wissenschaftlichen Konstruktionslehre*, Springer, Berlin, Heidelberg, Germany.

[15] Ponn, J. (2016), "*Systematisierung des Lösungsraums*", Lindemann, U. (Ed.), Handbuch Produktentwicklung, Carl Hanser, Munich, Germany, pp. 715-742.

[16] Gembarski, P.C., Sauthoff, B., Brockmöller, T. and Lachmayer, R. (2016), "*Operationalization of Manufacturing Restrictions for CAD and KBE-Systems*", Proceedings of the DESIGN 2016 14th International Design Conference, The Design Society.

[17] Ponn, J. and Lindemann, U. (2011), *Konzeptentwicklung und Gestaltung technischer Produkte: systematisch von Anforderungen zu Konzepten und Gestaltlösungen*, Springer, Berlin, Heidelberg, Germany.

[18] VDI (1993), *VDI Guideline 2221 - Methodology for developing and designing technical systems and products*, Beuth, Berlin, Germany.

[19] Ullman, D.G. (2009), *The mechanical design process,* McGraw Hill Higher Education, New York City, United States of America.

[20] Steiner, F. (2014), *Solution space development for mass customization: Impact of continuous product change on production ramp-up*, Verlag Dr. Kovac, Hamburg, Germany.

[21] Suh, N.P. (2005), *Complexity: theory and applications*, Oxford University Press on Demand, Oxford, United Kingdom.

[22] Shannon, C.E. (2001), "*A mathematical theory of communication*", ACM SIGMOBILE Mobile Computing and Communications Review, Vol. 5, No. 1, pp. 3-55.

[23] Jiao, J., Simpson, T.W. and Siddique, Z. (2007), "*Product family design and platform-based product development: A state-of-the-art review*", Journal of Intelligent Manufacturing, Vol. 18, No. 1, pp. 5-29.

[24] Aldanondo, M. and Vareilles, E. (2008), "*Configuration for mass customization: how to extend product configuration towards requirements and process configuration*", Journal of Intelligent Manufacturing, Vol. 19, No. 5, pp. 521 535.

[25] Ericsson, A. and Erixon, G. (1999), *Controlling design variants: modular product platforms,* Society of Manufacturing Engineers, ASME international, New York, United States of America.

[26] Schuh, G. (2005), *Produktkomplexität managen*, Carl Hanser, Munich, Germany.

[27] Lenders, M.J.E. (2009), *Beschleunigung der Produktentwicklung durch Lösungsraum-Management*, Apprimus, Aachen, Germany.

[28] Lüdtke, B. (2016), *Lösungsraum-Steuerung in der Produktentwicklung*, Apprimus, Aachen, Germany.

[29] Gero, J.S. (1990), "*Design prototypes: a knowledge representation schema for design*", AI magazine, Vol. 11, No. 4, pp. 26-36.

[30] Cox, J.J. (2000), "*Product Templates - A Parametric Approach to Mass Customization*", CAD Tools and Algorithms for Product Design. Springer, Berlin, Heidelberg, Germany, pp. 3-15.

[31] Gembarski, P.C., and Lachmayer, R. (2014), "*Forward variance planning and modeling of multi-variant products*", Procedia CIRP Vol. 21, pp. 81-86.

[32] Hermans, G. (2012), "*A model for evaluating the solution space of mass customization toolkits*", International Journal of Industrial Engineering and Management, Vol. 3, No. 4, pp. 205–214.

[33] Pahl, G. and Beitz, W. (2013), *Engineering design: a systematic approach*, Springer, Berlin, Heidelberg, Germany.

[34] Schlingheider, J. (1994), *Methodik zur Entwicklung rechnerunterstützter Konfigurationssysteme*, Carl Hanser, Munich, Germany.
[35] Huang, G.Q. (1996), *Design for X - Concurrent engineering imperatives*, Springer, Berlin, Heidelberg, Germany.
[36] Weber, C. (2007), "*Looking at "DFX" and "Product Maturity" from the Perspective of a New Approach to Modelling Product Development Processes*", Krause, F.-L. (Ed.), The Future of Product Development - Proceedings of the 17th CIRP Design Conference, Springer, Berlin, Heidelberg, Germany, pp 85-104.
[37] VDI (2009), *VDI Guideline 2209 - 3D Product Modelling*, Beuth, Berlin, Germany.
[38] Stroud, I. and Nagy, H. (2011), *Solid modelling and CAD systems: how to survive a CAD system*, Springer, Berlin, Heidelberg, Germany.
[39] Milton, N.R. (2008), *Knowledge Technologies*, Polimetrica sas, Monza, Italy.
[40] Chakrabarti, A., Shea, K., Stone, R., Cagan, J., Campbell, M., Hernandez, N.V. and Wood, K.L. (2011), *"Computer-based design synthesis research: an overview*", Journal of Computing and Information Science in Engineering, Vol. 11, No. 2, pp. 1-10
[41] Hoisl, F. (2012), *Visual, interactive 3D spatial grammars in CAD for computational design synthesis*, Ph.D. dissertation, Technical University Munich, Germany.
[42] La Rocca, G. (2012), "*Knowledge based engineering: Between AI and CAD. Review of a language based technology to support engineering design*", Advanced Engineering Informatics, Vol. 26, No. 2, pp. 159-179.
[43] McDermott, J. (1982), "*R1: A rule-based configurer of computer systems*", Artificial intelligence Vol. 19, No. 1, pp. 39-88.
[44] Sabin, D. and Weigel, R. (1998), "*Product configuration frameworks - a survey*", IEEE intelligent systems, Vol. 13,No. 4, pp. 42-49.
[45] Heinrich, M. and Jüngst, E.W. (1991), "*A resource-based paradigm for the configuring of technical systems from modular components*", Artificial Intelligence Applications, pp. 257-264.
[46] Richter, M.M. and Weber, R.O. (2016), *Case-based reasoning*, Springer, Berlin, Heidelberg, Germany.
[47] Gembarski, P.C. and Lachmayer, R. (2015), "*Degrees of Customization and Sales Support Systems - Enablers to Sustainability in Mass Customization*", Proceedings of the 20th International Conference on Engineering Design (ICED 15) Vol. 1: Design for Life, Milan, Italy, pp. 1-8.
[48] Sandrin, E., Trentin, A. and Forza, C. (2014), "Organizing for mass customization: literature review and research agenda", International Journal of Industrial Engineering and Management, Vol. 5, No. 4, pp. 159-167.

Razvoj prostora rešenja: Konceptualna razmatranja i razvoj matrice prostornih parametara kao alata za planiranje prostora rešenja zasnovanih na geometriji

Paul Christoph Gembarski, Roland Lachmayer

Primljen (18.10.2018.); Recenziran (21.11.2018.); Prihvaćen (03.12.2018.)

Apstrakt

Današnji CAD sistemi nude mogućnost modelovanja prostora rešenja zasnovanog na geometriji na osnovu tehnologije parametara i karakteristika. Ovde je prostor rešenja skup svih mogućih alternativa proizvoda iz kojih se, za definisani skup zahteva, mogu konfigurisati različite opcije. Neophodan korak pre modelovanja prostora rešenja je sticanje znanja o zavisnosti zahteva i rešenjima i ograničenjima koje diktira lanac nabavke, npr. ograničenja proizvodnje. U ovom radu, autori doprinose ovom polju razvijanjem matrice prostornih parametara (ParSM) kao alata za struktuirano izvlačenje zahteva, ograničenja prostora rešenja i rezultujućih parametara za CAD model. Nadalje, primena ParSM-a je prikazana i objašnjena na tosteru sa promenjivim elementima kućišta gde su proizvodna ograničenja rezultat procesa aditivne proizvodnje.

Ključne reči: *Planiranje parametara, matrica prostornih parametara, razvoj prostora rešenja, konfiguracija proizvoda*