

8th International Conference on Mass Customization and Personalization – Community of Europe (MCP-CE 2018)

Digital Customer Experience September 19-21, 2018, Novi Sad, Serbia



THE PARAMETER SPACE MATRIX AS PLANNING TOOL FOR GEOMETRY-BASED SOLUTION SPACES

Paul Christoph Gembarski, Roland Lachmayer

Institute of Product Development, Leibniz Universität Hannover, Welfengarten 1A, 30167 Hannover,

Germany

Abstract: Based on parametrics and feature technology, today's CAD-systems offer the possibility to model geometry-based solution spaces from which a product 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 the following paper, the authors contribute to this by development of the Parameter Space Matrix (ParSM) as a tool for a structured elicitation of such dependencies. As such, it allows planning the parameters in a CAD-model. 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. The end customer then can modify these body elements according to his own aesthetic requirements in a co-design process in a configurator.

Key Words: Co-Creation, Product Configuration, Parameter Planning, Parameter Space Matrix, Solution Space Development

1. INTRODUCTION

For about 20 years, the use of computer-aided design (CAD) and engineering (CAE) tools has backed the steady increase in competitiveness and innovation of many companies. In particular, the use of parametric design systems, in which not only the shape of a component or an assembly is modeled, but also their describing parameters, leads to great potentials in adaptive and variant design [1].

Especially, the possibility of defining mathematical and logical constraints between parameters in a CAD system makes it possible to implement knowledge in explicit form within digital prototypes. As a result, configuration and derived parameters can be distinguished from each other. In relation to variant design, the designer does not only specify the product shape but also the control and configuration concept for his component and thus describes a solution space [2, 3].

1.1. Motivation and Aim

However, looking at the use of the tools in the operational environment and its practical implication more in detail, the need for a new variant or changes to the product lead to ever new CAD and CAE models. The CAD model itself is usually created with regard to a single product variant, not with respect to a solution space, which also covers possible (future) variants. 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 [4-6].

Considering suppliers where a mass customization business model is the foundation of entrepreneurial activity, the design of the solution space is a key principle [7]. The ability to cope with the resulting complexity is enabled by knowledge-based engineering (KBE) systems in general and product configurators in particular [3, 8-9]. Process models for creating KBEapplications and prior to that acquiring the necessary knowledge to be implemented are available (refer e.g. to [10, 11]). But although contemporary CAD-systems offer the possibility to implement knowledge in the digital prototypes themselves (e.g. refer to [12]), there is a lack of concrete modelling principles or detailed application examples. For closing a part of this gap, the authors developed the parameter space matrix (in the following ParSM) as a parameter and constraint planning tool for geometry based solution spaces.

1.2. Structure of the Paper

In the following Section 2, the theoretical background of design solution spaces is reflected from literature. In Section 3, setup and functioning of ParSM is described. Section 4 then refers to the use of ParSM in the case study "customizable toaster". Subsequently in section 5, the implications of the case study on parametric CAD design in general is discussed and how ParSM supports the planning of a component's parameters. The final section 6 summarizes the article and presents further research potentials.

2. THEORETICAL BACKGROUND

In order to get a fundamental understanding of solution spaces as a basis for creating a planning aid for solution space development, a structured literature review was performed that is exemplarily presented in the following sub-section 2.1. Afterwards in sub-section 2.2, the state of the art in CAD-based solution space modelling is presented.

2.1. Design Solution Spaces

Briefly, the literature review on solution space development showed four different views on design solution spaces that could be consolidated and are discussed in detail in the following:

- 1. External variety view: The solution space as a set of all existing product variants that either fulfill specific functions or defined requirements (refer e.g. to [13-15])
- 2. Internal variety view: Solution spaces as a set of all existing product components and the lawsof-creation, how these are connected to the final product (refer e.g. to [16, 17])
- 3. Exploration view: The solution space as a search space for an artifact to be developed, which is converged as the development process progresses (refer e.g. to [18-20])
- 4. Degree-of-Freedom view: The solution space as a variation space of a known design in the sense of a design prototype (refer e.g. to [21, 22])

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 [13]. 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 [14]. In the product development process, the required product properties are compared with the properties of the designed artifact and approximated by synthesis analysis loops (fig. 1).



Fig. 1. Requirement space and solution space in the *development process* (adapted acc. to [14])

Usually, not all areas of the solution space are accessible during the development process. The formulation of restrictions is thus a tool for limiting the design space early in the design project [23]. Restrictions originate from various circumstances: The most practical implication results from manufacturing processes where manufacturing restrictions (i.e. travelling distances of milling machines, effective hardening depths, etc.) are coded within design guidelines. In product development this is considered in the so called design for X-approaches [e.g. 24-26]. Additionally, restrictions from legislation (e.g. abdication of combustion engines or pollution thresholds) and economic restrictions (available resources in product development) have to be reflected [26, 27].

Nonetheless, most approaches that cover this **external variety view** on a solution space include interactions and connections between solution space and requirement space only on conceptual levels or with reference to requirement engineering.



[16]) [16])

Internal Variety View. Such relationships are integral part of the design method Axiomatic Design [16]. 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).

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). 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.



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

If alternative design matrices exist, the information axiom states that the one design with the lowest information content should be favored. The information content's calculation is grounded on Shannon's information-based entropy and is considered a measure of structural complexity in information technology [28].

Another domain concept as description for solution spaces is proposed by Aldanondo [17]. 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). time. Aldanondo explicitly focuses 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 is joint product and production process configuration.

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.

Exploration View. Basically, authors that describe approaches for solution space exploration use a mix of external and internal variety view. E.g. Lenders [18] and Lüdtke [20] 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 corridors instead of requirements. 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 artifact 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.



Fig. 4. Combined Product and Process Configuration acc. to [17]

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



Fig. 5. Design Prototypes (acc. to [21])

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 [21] 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). 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].

2.2. CAD-based Solution Space Modelling

The basis for modelling a geometry-based solution space is the design system's ability to differentiate between shape and its describing parameters [2]. In parametric CAD, parameters can be related by mathematical and logical constraints that establish editable equation systems [12]. Furthermore, chronology-based references determine the genesis of the model and thus the sequence of all the individual operations for geometry creation and modification [29]. By defining such dependencies and user-defined parameters, it is possible to explicitly implement design knowledge in a CAD model [30].



Fig. 6. Overview of the principles of 3D modelling [29]

The German VDI guideline 2209 [29] mentions two other types of CAD systems, which provide additional functionality for creating variable geometry models and for mapping design knowledge (fig. 6). Feature-based CAD systems 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 behavioral rules [5]. 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 'knowledgebased 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' [5].

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 [31]. This domain knowledge may be expressed e.g. by dimensioning formulae that constrain parameters of the CAD-model. Other ways of formalizing domain knowledge is templates that have to be understood as reusable, updatable building blocks in a virtual prototype [22]. Geometric templates may correspond to Gero's routine or innovative design activity as these provide variable geometry and configuration parameters. Such a template can be used as starting point for detailed design respectively.



Fig. 7. Knowledge Modelling in KBE and KBD

Second, control knowledge determines the way a solution space is explored. A possible way to do this is rule-based reasoning. Rules are if-then-else-statements that are fired procedurally. Although it's one directional and simple nature, instantiation and loops form complex rule bases where rules activate sub-ordinate rules or exclude them from further processing [32]. For a detailed overview of knowledge modelling techniques in contemporary CAD-systems refer to [12].

3. DEVELOPMENT OF THE PARAMETER SPACE MATRIX

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 predeccessor 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 allready known and determined. Then, the design task is to flexibilize 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 geometry to the later design parameters like introducted in axiomatic design without violating any constraints. So, the design task is to determine the relationships of requirements, model parameters and restrictions.

R	EQU	IREN	/IENT	S						REST	RICT	IONS	5
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		MOD	EL PAF <u>BAS</u>	RAMET E PLAT	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	Ν	2500	Applied Force					
	4				P:f	mm	4	Maximum Deformation					
		х	х		P:L	mm	600	Bounding Box Length			< 800	< 720	
					P:L_AB	mm	220	Length AB to Connector	< 225	< 250			
		х	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			

Fig. 8. Parameter Space Matrix

Since until now no computer-aided tools for the joint elicitation of requirements, parameter hierarchies and restrictions exist, the authors invented 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 (see sub-section 3.4) should be available.

Basically, ParSM is an extended parameter table of the later CAD-model (fig. 8). Depending on the CADsystem, ParSM can be created within the CAD environment or in a spreadsheet application. The following description bases on an Excel macro spreadsheet.

3.1. Center Part: Parameter Table

The central element of the matrix is the parameter list of the component. 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.

NAME	UNIT	=Buildspace X - 2*Wall Thickness - Installation Space - 80						
P.TS	mm		··· Wall Thickness					
S:L1	mm	280	Buildspace X					
S:L2	mm	60	Installation Space					
P:L_s	mm	128	Bounding Box Depth					

Fig. 9. Parameter Constraining

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 occurence 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 CADsystems only a limited count of mathematical operators is available. The use of a spreadsheet application extends this.

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 destinguish 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).

3.2. Left Wing: Requirement Specification

To the left of the parameters are the requirements and their influence on the model parameters. 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 priortized 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, either 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.

3.3. Right Wing: Restriction List

The right part of ParSM is dedicated to the restrictions. 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 favorable use of existing jigs and clamping tools.

3.4. Conflict Detection and Resolution

Bookmarking requirements, model parameters and restrictions in one common matrix allows also detecting and resolving conflicts. The most simple 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.



Fig. 10. Modular adaptable Toaster Housing

ParSM has been used amongst other projects for the planning of a 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 color can be chosen from a given list since the processed parts are dip-coated. Some example configurations are depicted in fig. 11.



Fig. 11. Housing Variations

Fig. 12 on the following page 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.

F	REQU	IREN	1ENT	S						RE	STRI	стіо	NS	
1.1	1.2	2.1	2.2						A1	A2		M1	M2	
Number of Slots	Max. Surface Temperature	Adaptable shape	Symmetry		MOD <u>(</u>	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		
х		x			P:e1_y1	mm	-120	Upper Plane Edge 1				480		
		х	х		P:e1_x2	mm	180	Upper Plane Edge 2				540		
х		x			P:e1_y2	mm	-120	Upper Plane Edge 2				480		
		x	х		P:e1_x3	mm	-180	Upper Plane Edge 3				540		
х		х			P:e1_y3	mm	120	Upper Plane Edge 3				480		
		x			P:e1_x4	mm	180	Upper Plane Edge 4				540		
х		x			P:e1_y4	mm	120	Upper Plane Edge 4				480		
		х			P:e1_H	mm	60	Upper Plane Height	60-80			280		
		х			P:e2_x1	mm	-180	Lower Plane Edge 1					18°	
х		х			P:e2_y1	mm	-120	Lower Plane Edge 1					18°	
		х			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					

Fig. 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.

5. DISCUSSION

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. 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, modeling 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. An example of this is the various skeletal techniques used for the respective CAD systems. 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. E.g. 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 then.

6. SUMMARY AND FURTHER RESEARCH

The present article discusses the state of the art of solution space development. Following the presentation of different views on design solution spaces, the parameter space matrix was derived as a planning aid for parameter planning and constraining in CAD models.

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. 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. 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.

7. REFERENCES

- [1] S. Vajna, *CAx für Ingenieure: eine praxisbezogene Einführung*, Berlin, Heidelberg, Germany: Springer, 2009.
- [2] J.J. Shah, "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, 1998, pp. 53-68, 2001.
- [3] P.C. Gembarski, R. Lachmayer, "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, 2017.
- [4] C. Weber, "Design Theory and Methodology-Contributions to the Computer Support of Product Development/Design Processes" *The Future of Design Methodology*. Berlin, Heidelberg, Germany: Springer, pp. 91-104, 2011.
- [5] M. Hirz, Integrated Computer-aided design in automotive development, Berlin, Heidelberg, Germany: Springer, 2013.
- [6] P.C. Gembarski, H. Li, R. Lachmayer, "Template-Based Modelling of Structural Components" *International Journal of Mechanical Engineering and Robotics Research*, vol. 6, no. 5, pp. 336-342, 2017.
- [7] F. Salvador, P.M. De Holan, F. Piller, "Cracking the code of mass customization" *MIT Sloan Management Review*, vol. 50, no. 3, pp. 71-78, 2009.

- [8] L. Hvam, N.H. Mortensen, J. Riis, *Product Customization*, Berlin, Heidelberg, Germany: Springer, 2008.
- [9] R. Reichwald, F. Piller, *Interaktive Wertschöpfung*, Wiesbaden, Germany: Gabler, 2009.
- [10] G. Schreiber, B. Wielinga, R. de Hoog, H. Akkermans, W. Van de Velde, "CommonKADS: A comprehensive methodology for KBS development" *IEEE expert*, vol. 9, no. 6, pp. 28-37, 1994.
- [11] M. Stokes, Managing engineering knowledge: MOKA: methodology for knowledge based engineering applications, London, Great Britain: Professional Engineering Publishing, 2001.
- [12] P. C. Gembarski, H. Li, R. Lachmayer, "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, 2015, pp. 265-283.
- [13] V. Hubka, Theorie Technischer Systeme: Grundlagen einer wissenschaftlichen Konstruktionslehre, Berlin, Heidelberg, Germany: Springer, 1984.
- [14] J. Ponn, "Systematisierung des Lösungsraums" Lindemann, Udo (ed.): Handbuch Produktentwicklung, Munich, Germany: Carl Hanser, pp. 715-742, 2016.
- [15] F. Steiner, Solution space development for mass customization: Impact of continuous product change on production ramp-up, Hamburg, Germany: Verlag Dr. Kovac, 2014.
- [16] N.P. Suh, Complexity: theory and applications. Oxford, United Kingdom: Oxford University Press on Demand, 2005.
- [17] M. Aldanondo, E. Vareilles, "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, 2008.
- [18] M.J.E. Lenders, Beschleunigung der Produktentwicklung durch Lösungsraum-Management, Aachen, Germany: Apprimus, 2009.
- [19] J. Ponn, U. Lindemann, Konzeptentwicklung und Gestaltung technischer Produkte: systematisch von Anforderungen zu Konzepten und Gestaltlösungen, Berlin, Heidelberg, Germany: Springer, 2011.
- [20] B. Lüdtke, Lösungsraum-Steuerung in der Produktentwicklung, Aachen, Germany : Apprimus, 2016.
- [21] J.S. Gero, "Design prototypes: a knowledge representation schema for design" *AI magazine*, vol. 11, no. 4, pp. 26-36, 1990.
- [22] J.J. Cox, "Product Templates A Parametric Approach to Mass Customization" *CAD Tools and Algorithms for Product Design*. Berlin, Heidelberg, Germany: Springer, pp. 3-15, 2000.
- [23] P.C. Gembarski, B. Sauthoff, T. Brockmöller, R. Lachmayer, "Operationalization of Manufacturing Restrictions for CAD and KBE-Systems", *Proceedings of the DESIGN 2016 14th International Design Conference*, The Design Society, 2016.

- [24] Huang, G.Q., "Design for X Concurrent engineering imperatives", Springer Science + Business Media Dordrecht, 1996.
- [25] Weber, C.: "Looking at "DFX" and "Product Maturity" from the Perspective of a New Approach to Modelling Product Development Processes", The Future of Product Development - Proceedings of the 17th CIRP Design Conference, Krause, F.-L. (ed), Springer-Verlag Berlin Heidelberg, pp 85-104, 2007.
- [26] D.G. Ullman, The mechanical design process. New York City, United States of America: McGraw Hill Higher Education, 2009.
- [27] G. Pahl, W. Beitz, *Engineering design: a systematic approach*, Berlin, Heidelberg, Germany: Springer, 2013.
- [28] C.E. Shannon, "A mathematical theory of communication" ACM SIGMOBILE Mobile Computing and Communications Review, vol. 5, no. 1, pp. 3-55, 2001.
- [29] VDI, VDI Guideline 2209 3D Product Modelling, Berlin, Germany: Beuth, 2009.
- [30] I. Stroud, H. Nagy, *Solid modelling and CAD* systems: how to survive a CAD system, Berlin, Heidelberg, Germany: Springer, 2011.
- [31] N.R. Milton, *Knowledge Technologies*, Monza, Italy: Polimetrica sas, 2008
- [32] G. La Rocca, "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, 2012.

CORRESPONDENCE



Dipl.-Ing. (FH) Paul Gembarski Institute of Product Development, Leibniz Universität Hannover, Welfengarten 1A 30167 Hannover, Germany <u>gembarski@ipeg.uni-hannover.de</u>



Prof. Dr.-Ing. Roland Lachmayer Institute of Product Development, Leibniz Universität Hannover, Welfengarten 1A 30167 Hannover, Germany <u>lachmayer@ipeg.uni-hannover.de</u>