ANNALS of Faculty Engineering Hunedoara
– International Journal of Engineering
Tome XV [2017] – Fascicule 1 [February]

ISSN: 1584-2665 [print; online]
ISSN: 1584-2673 [CD-Rom; online]
a free-access multidisciplinary publication
of the Faculty of Engineering Hunedoara

[1.]Paul Christoph GEMBARSKI, [2.]Roland LACHMAYER

HOW RULE-BASED SYSTEMS IMPACT PRODUCT COMPLEXITY

[1-2.] Institute of Product Development, Leibniz Universität Hannover, Welfengarten 1A, Hannover, GERMANY

Abstract: Managing complexity in product development is one of the side effects of knowledge-based-engineering (KBE). KBE systems allow the automation of design synthesis tasks as well as product configuration, like used in mass customization business contexts. Here, rule-based systems belong to the oldest but still used implementations of KBE-methods. In the present article it is discussed how rule-based systems impact product complexity. Therefore, four different complexity measures are developed and visualized in the Hannover House of Complexity which has to be understood as framework for company specific complexity management. It focusses on size, degree of exploration and closure of the design solution space and the connectivity of multiple solution spaces. Afterwards, rule-based systems are characterized and assessed at the examples of R1/XCON and MYCIN as well as actual implementations within CAD-systems and configurators.

Keywords: Product Complexity, Product Variety, Hannover House of Complexity, Rule-Based Systems, Knowledge-Based-Engineering

INTRODUCTION

Over the last years, increasing demands on performance and quality of new or enhanced products lead to shorter product lifecycles and hence shorter development times. Other important constraints in product development are the globalization of supply and demand as well as a growing tendency towards product customization. These boundary conditions lead to a frequent adaptation of products to different functional or design requirements and so to larger product variety [1].

The resulting complexity in product development due to the aforementioned issues is approached by the methods of variant design like parametric designs, type series, modular design kits or design platforms [2].

In order to explore the so defined solution space rapidly and efficiently as well as to ensure a high level of innovativeness, the utilization of existing design knowledge and the automation of routine design tasks are critical success factors. So, the organizational efforts for creating product variety are minimized [3].

The term complexity is often used synonymously for product variety in this context. A generally accepted definition for complexity in engineering design is yet not at hand but most approaches include organizational effects and take into account that high variety leads to problems and uncertainties in forecasting demands and in control of manufacturing and operations. Furthermore, complexity is considered to be strongly company specific [4].

Different approaches for complexity management exist which target e.g. on mastering product variety or production complexity. The Hannover House of Complexity is a more general framework where business typology and complexity measures as well as methods and tools for complexity management are joined [5].

In the present article, rule-based systems as one example of knowledge-based (KB) or knowledge-based-engineering (KBE) applications are assessed regarding their impact on product complexity.

Motivation

Especially in the competitive strategy of mass customization, the resulting need for flexibility in product development and manufacturing calls for adequate information technology support. Solution space development using product configuration systems is considered as one building block for complexity management.

Product configuration systems belong to the field of KB and KBE applications. From point of view of computer-aided engineering, KBE extends the abilities of parametric modeling by implementing explicit design knowledge into the virtual product model [6].

Rule-based systems belong to the oldest but still deployed applications of KB/KBE. Used as reasoning mechanism in the early expert systems in the 1980ies and 1990ies they provided sales support as configuration systems and they automated single routine tasks and decision-making processes in various disciplines of engineering design. Today, many CAD-systems still have the possibilities to use design rules for variant design automation.

Nevertheless, the impact of rule-based systems on product complexity and solution space development is still an open question. This article aims at bridging this gap and discusses how such systems impact product complexity measures

Structure of the Article

In the following section 2, a brief introduction into the concept of product complexity, its measures and its management is given. Afterwards in section 3, the Hannover House of Complexity is introduced as complexity management framework. Section 4 contains the discussion of rule-based systems. Their assessment regarding the single complexity dimensions is part of section 5. The final section 6 summarizes the article and draws further research questions.

PRODUCT COMPLEXITY

Generally, cybernetics and system theory are considered as origin of complexity theory [7]. From these, different approaches have been derived and further developed for various scientific disciplines such as natural science, social or labor science [8]. Nonetheless, general definitions or modeling principles do not exist. Complexity is rather mapped and reduced on the particular problem statement.

Approaches in engineering design are typically broken down to complexity of products as well as development and production processes. Usually, external and internal product complexity is differentiated. The first is understood as diversity of a company's offering (number of product variants), the latter is defined as number of subassemblies and components as well as combination rules and design knowledge about assembling these components to end products [9].

A lot of authors emphasize that product complexity and process complexity are strongly intertwined. Multi-variant products thus lead to an increase of complexity in all operational structures and processes because the high quantity of end products and their components as well as the corresponding documents for each project and each customer have to be managed in operations and the whole supply network [10].

Complexity Measures

Prerequisite for the management of complexity, it is necessary to determine an ideal amount of complexity or to differentiate between good and bad complexity. The early attempts of finding such descriptive dimensions failed and resulted in a multitude of measures which indeed could not exactly assess complexity [4].

For his own complexity management approach, Schuh uses the so called complexity drivers which is diversity on the one hand and dynamics on the other hand. His concept of diversity encompasses both the diversity of system elements and the diversity of relations between those elements as well as the variety of system states over time [11].

Gießmann uses a compact approach from point of view of logistics and describes complexity in the dimensions of variety, heterogeneity, diversity and uncertainty. All these dimensions are dependent since e.g. an increase of dynamics results in an increase of uncertainty because the prediction of future developments and system states is more difficult. So, it is not enough to measure a single aspect of complexity or to consider only a limited count of system elements but to examine the whole system and all possible occurrences [12].

Broken down to manufacturing organizations, Frizelle reduces this to even two dimensions by the consideration that complexity arises out of the presence of variety. Increasing variety generates uncertainty so that the system's behavior cannot be completely predicted. According to Frizelle, "variety can be seen in terms of trajectories – the path a system traces over time; the greater the variety, the more trajectories are open to the system. Uncertainty comes from not knowing which trajectory the system will follow" [4].

When configurable products for e.g. mass customization are designed, the possible product complexity is reflected by the solution space from which the individual variant is set up from. Here, diversity and uncertainty are both concepts that can be used for assessment of product complexity and thus lead to four complexity measures (fig. 1):

- Size of the possible solution space: How many product variants / possible solutions are described in the solution space?
- Closure of the possible solution space: Are all limitations, either technical like manufacturing restrictions, design interfaces or economic ones such as minimum lot sizes, etc., known and formulated explicitly?
- Degree-of-Exploration: Are all product variants predefined or pre-calculated or are there any unknown areas?
- Connectivity of multiple solution spaces: How many solution spaces interact with each other in which way?

Linked to product development, three observations stand out. First, the degree-of-exploration also marks a potential for conflicting solution elements. Since most product designs for e.g. mass customization rely either on parametrization or on aggregation of predefined components, restrictions of value ranges or combinations of options are usual to obviate unfeasible designs. So, if not all possible end product variants are predetermined, the validity of some variants may not be checked beforehand unless all relations between all components and all restrictions are modeled explicitly.

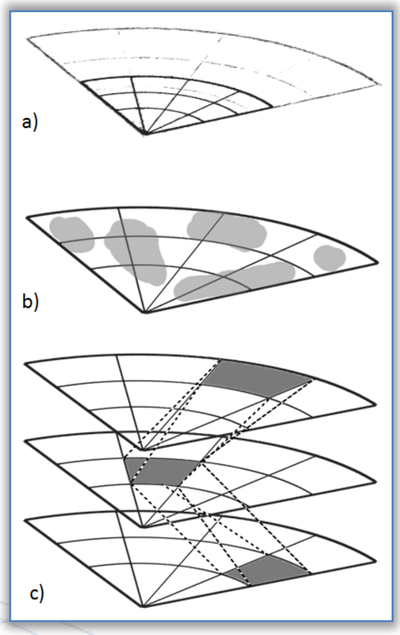


Figure 1. Complexity Measures of a Solution Space for Product Development: a) Size and closure of the Solution Space, b) Degree of Exploration of the Solution Space, c) Connectivity of multiple Solution Spaces

Secondly, commonalities are expressed in the connectivity of multiple solution spaces which is important for change management. Interacting solution spaces mirror parent-child relations on the one hand, since the solution space of the end product is linked to the solution space of the constituting sub-assemblies and so forth. On the other hand they show cross-references where a sub assembly is built into different end products. Interacting solution spaces are linked via constraints which correspond to functional or logical relations. The more relations a system has, the more complicated is the prediction of effects when components change.

Third, the size of a solution space is hard to calculate if it is not fully explored and the design consists of multiple variable components. So both, the size and the degree of exploration, might rely on estimates and so be uncertain themselves.

Complexity Management

According to Schuh, the management of complexity is “the design, development and control of business activities regarding products, processes and resources. Managing complexity aims at dominating diversity along the whole value chain, so that customer satisfaction as well as organizational efficiency gets maximal” [11].

Although many authors point out, that a certain amount of complexity may be beneficial to a company since heterogeneous market demands may only be satisfied through variety, most of the management processes found in literature aim singularly at reducing complexity. A planned development of complexity as a capability is discussed rarely.

Nonetheless, different aspects of complexity management and single methods can be found in literature. Bliss concludes that the major process management schools of the 1990’s (i.e. lean management, business process reengineering and variant management) may also be considered as complexity management methods. Especially variant management concentrates efforts on product complexity and customer complexity, i.e. the breadth and heterogeneity of the customer base [13]. Here, e.g. modularization is a valuable building block.

From this point of view, Bliss’ argumentation leads to three very fundamental fields of complexity management [5]:

- *Management of product complexity*, i.e. measures in different areas of the company, which purpose is designing and controlling the complexity of end products as well as their components and individual parts depending on their functional and design requirements.

- *Management of resource complexity*, i.e. methods in order to design and control the complexity of production resources, raw materials as well as knowledge and personnel in the value chain.
- *Management of process complexity*, i.e. approaches that aim at design and control of complexity of operational and organizational structures.

When a high degree of complexity already exists, three basic courses of action should be carried out. First, existing complexity has to be reduced which targets at streamlining the existing product and process portfolio for a short-term effect on product complexity. As result, product variants with low demand and overlaps in the over-all offering are identified and then eliminated.

Secondly, the implementation of complexity control aims at strategic planning and development of the necessary complexity. Here, the methods of variant design like product family design, modular design kits and solution space modeling in general are subsumed. Additionally, an according setup of the manufacturing organization and of order processing has to be implemented.

The last step is prevention of complexity. All new product and process variants have to be assessed regarding additional benefits for company and customer before realization and implementation.

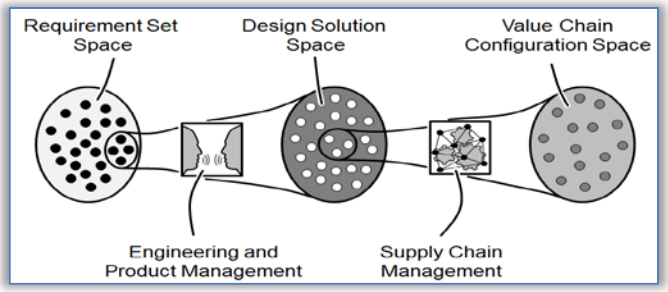


Figure 2. Relation of Requirement Set Space, Design Solution Space and Value Chain Configuration Space

If complexity has to be developed purposefully, the available solution space must be designed in advance. Therefore, on the one hand, the translation process of requirements towards a specific product specification has to be automated. On the other hand the capabilities of the supply network or the production processes (which we understand as value chain configuration space, fig. 2) must be formulated as restrictions for the solution space. This is necessary especially for large solution spaces because not all end product variants might be defined beforehand, but their constituents and creation principles.

Afterwards, every change in the supply network and in production leads to a verification of the existing solution space restrictions and borders. The solution space may be continuously extended by newly developed product variants, e.g. if a new key technology is introduced, as long as it corresponds to the available value chain configuration space.

HANNOVER HOUSE OF COMPLEXITY

The Hannover House of Complexity is a framework in which different methods, tools, etc. are classified with regard to their effect on the single complexity measures. Fig. 3 shows the basic architecture. In principle, the design is similar to the House of Quality known from Quality Function Deployment. In opposite to QFD, the major areas are not the mapping of customer requirements to functions or properties of the product but the mapping of different building blocks for complexity management and their particular effects on different complexity dimensions. In the roof of the House of Complexity the interdependencies between these building blocks are rated to estimate weather two of these building blocks intensify their benefit or extenuate each other. Since the framework is setup as aid for decision-making, a reference to a standard company of an according business type is given for comparison. This includes the choice of typical building blocks on the one hand. On the other hand it also allows the assessment of the usual complexity profile at this particular business type. The architecture of the House of Complexity is completed by the fields for the as-is-analysis. An example of the detailed framework is given in fig.4.

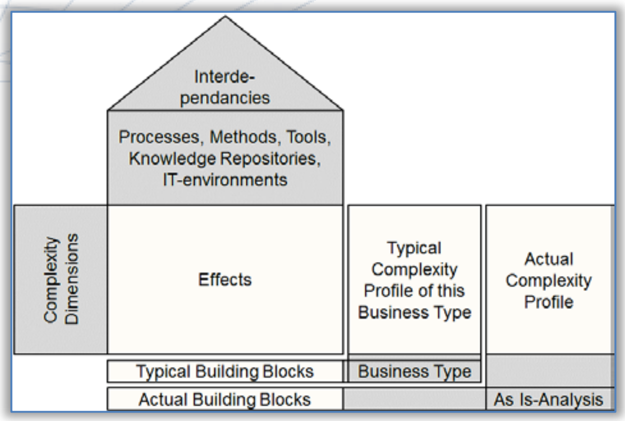


Figure 3. The Hannover House of Complexity – Architecture [5]

In the example, the effect of different building blocks for complexity management on the dimensions of product complexity is shown conceptually. Based on a business typology a company assigns itself to a business type 1. Comparing both complexity profiles shows that in contrast to the benchmark the connectivity of solution spaces and the closure differ. This is due to the missing of a complexity management building block which is yet not implemented at the company.

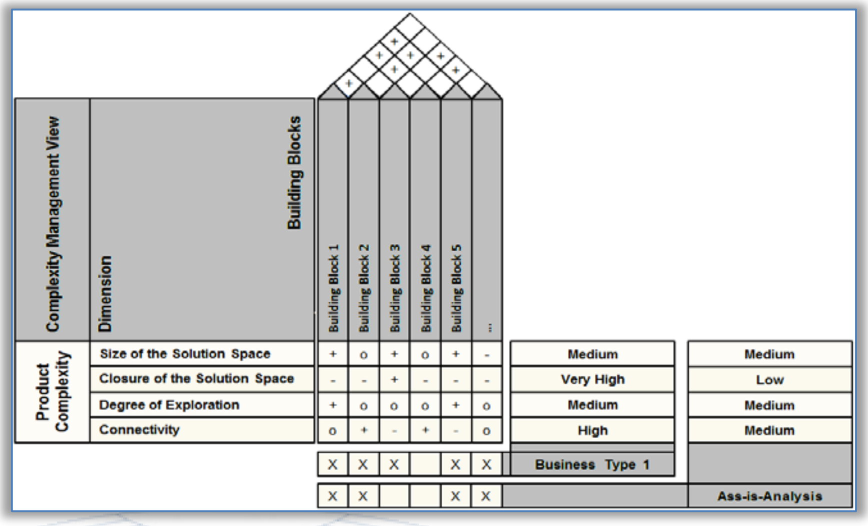


Figure 4. The Hannover House of Complexity – Framework [acc. to 5]

Furthermore, the roof of the House of Complexity depicts the mutual effects of building blocks one to five. As can be seen from this example, it is not the aim of minimizing every complexity dimension. In the example above, the uncertainty of the systems behavior increases.

RULE-BASED SYSTEMS

From a wider angle, the early rule-based systems belong to the class of knowledge-based systems which purpose was to replicate human experts for certain problem solving domains. This can generally be divided into two blocks. The first deals with synthesis (e.g. synthetic design, configuration or planning), the second one targets at analysis (e.g. classification, diagnosis or prediction) [14]. In the next sub-section, two examples of expert systems form the 1980ies are presented where R1/XCON represents a system for design synthesis and configuration while MYCIN stands basically for a diagnosis system.

The second sub-section shifts the focus from knowledge-based to knowledge-based-engineering systems. Here, problem-solving is linked directly to computer-aided-design [15].

In both application domains, rules show up as approach for knowledge representation as well as inferencing. Basically, the rule concept is grounded upon the IF-THEN-ELSE-notation known from software development. The tools for creating rule-based systems are easy to learn and simple to use. Nevertheless, different authors point out that such systems become difficult to maintain when they grow very large and reach a certain amount of rules [16].

As stated by Cederfeldt, rules are able to code the following categories of knowledge and problem solving abilities [17]:

- Purely empirical knowledge: Statements of facts and relations derived from experiments. This type of knowledge is usually of explicit kind.
- Rules of thumb / common practice / heuristics: Simplified statements of facts and relations derived from experience. Heuristics are formulated explicitly or implicitly.
- Common Sense: Statements about beliefs or habits derived from e.g. tradition or personal perspective. That kind of knowledge is usually implemented as implicit or explicit network of information.
- Logic Reasoning: Ability to conclude effects or actions from rules and facts. In context of KB or KBE this has to be stated based on explicit knowledge.

Rule-Based Systems in the early Days of KBE

One of the most famous and discussed implementations of a rule-based system is McDermott's R1/XCON configurator. It was designed to configure VAX-11/780 computer systems and proved valuable support for the sales department because the validity of each requested variant was checked immediately based on the customer order. If the configurator identified any incompatibilities it could provide assistance in modifying the design according to the given requirements [18].

Knowledge had to be represented in two different contexts. On the one hand knowledge about the available sub-components of a VAX-11 computer system was hardcoded, i.e. electrical properties, number of interfaces to other sub-components, etc.

On the other hand, rules had to be implemented that allow the formulation of feasible designs. Therefore, knowledge about constraints in the system configuration must be formalized in an explicit way (e.g. if the number of data storages exceeds the controller capacities, the configurator must either warn the user or give him advice to choose a controller with more ports), as well as associations of sub-components (if the one is chosen, the depended one has to be chosen as well).

The system was a classic procedural program in which the configuration task was traversed in a sequential way. Nevertheless, not all rules were fired since the system had the ability of deleting unnecessary rules from the working memory or including new sets of rules where needed for decision-making. Therefore, so called sequencing rules were used which determine the order in which decisions in the configuration process have to be made so that the resulting end product variant is valid. Started with over 770 rules and approximately 300 components the system developed over its life time to 17.500 rules and over 31.000 components. Due to product development, nearly 40 percent of all rules had to be revised yearly.

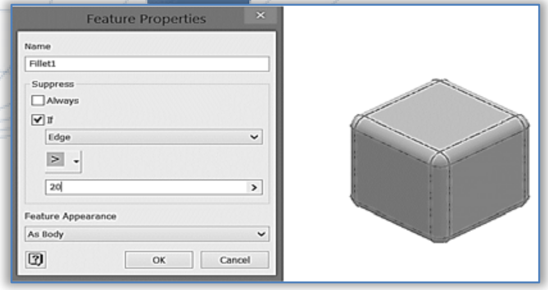
Another original implementation of a rule-based system is MYCIN which was planned as diagnosis system for infectious diseases. Here, the rule concept was used particularly due to its ability to capture heuristic knowledge (rules of thumb).

In contrast to R1/XCON, MYCIN was designed to explain its reasoning to the user. In that special case, the rule base has to be understood as network of goals (analysis of the patient's state or advice for medical treatment), hypothesis (possible causes for the patient's state) and the constraining rules [19]. Besides the formulation of explicit knowledge in rules, MYCIN shows a crisp separation of domain and control knowledge. The first is called structural knowledge and holds the knowledge about problem features and diagnosis. The latter is called strategic knowledge and is represented by meta-rules that order and restrict rule activation and reasoning.

Rule-Based Systems today

Especially for local and well-structured problem domains the rule concept is still state-of-the-art. Many of today's commercial knowledge-based configuration systems still use the rule concept with stronger or minor focus. An example is web-configuration in automotive development where a lot of sales configurators are set-up on a more or less procedural decision tree (at first choose the car model and then decide for an appropriate engine and gear, etc.).

Another field of use is e.g. the domain of product-service-system configuration. A product-service-system is considered as offering, where physical and non-physical components like services are developed co-equally and in an integrated way in order to provide certain functionalities for the customer. Here, the lack of a common data model for product and service components calls for rule-based configuration systems since rules are able to express the causal and logical relationship between them without further effort [20].



Feature Properties

Name

Fillet1

Suppress

☐ Always

☑ If

Edge

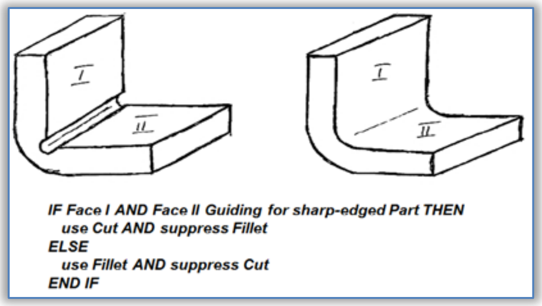
>

20

Feature Appearance

As Body

Figure 5. Suppression state definition in feature properties dialog



```
IF Face I AND Face II Guiding for sharp-edged Part THEN
   use Cut AND suppress Fillet
ELSE
   use Fillet AND suppress Cut
END IF
```

Figure 6. Definition of shape feature alternatives for extrusion profiles via design rule

Also in the domain of knowledge-based-engineering the rule concept is widely used. In contrast to the aforementioned knowledge-based systems, KBE aims commonly at the modification or analysis of a geometric product model which is available in a computer-aided engineering, especially computer-aided design system. On the one hand, many CAD-systems have the ability of using design rules directly in product modeling [21].

As an example, Autodesk Inventor Professional uses two different rule implementations. First, within the part modeling environment, the suppression state of a feature and a parameter may be linked via rules. In the example shown in fig. 5 the cube’s fillet is suppressed when the length of the edge (described in a parameter named edge) exceeds 20 mm.

Another way of defining rules is the iLogic environment. The iLogic programming language is similar to script languages. Common constructs like if-then-else or select-case decision trees, while loops, the use of sub procedures and a class concept are usable. As command library the snippets include code templates for almost every modeling context within Inventor.

A use-case in this context is the formulation of manufacturing restrictions where rules are used to express explicit design knowledge that has a local influence on the surrounding geometry. E.g., when it is necessary to enclose a sharp-edged component within a hollow profile in extrusion molding the edge of the profile cannot be rounded as it is recommended. So, the rule can be formulated as depicted in fig. 6.

On the other hand, design support systems for adjacent design activities like manufacturing process design, tooling or fixture design embed rules. As example, Xuewen describes such a system for hammer forging design which was implemented as add-in for SolidWorks. Here, the rule-base is only one knowledge representation which is coupled with model-based approaches [22]. Hunter Alarcón synthesized a system for fixture design where the use of heuristic knowledge is similar compared to MYCIN. The system consists of a catalogue of standard parts for fixture design, an analysis system for the geometry of the machined part, multiple sets of rules for functional and detailed design and a model-base for functions and machining processes [23].

COMPLEXITY EFFECTS OF RULE-BASED SYSTEMS

For the analysis of the impact on product complexity, several systems form literature as well as self-developed KBE-Systems were assessed regarding the aforementioned complexity measures. The analysis leads to the following assumptions:

- Size of the possible solution space: When rules are used, the existing solution space is not affected regarding its size. When applied like in R1/XCON, rules check for the consistency of the targeted solution but generally do not invent new ones. When applied like in Hunter Alarcón’s fixture design system, the rules lead to predetermined solutions which had been encoded in rules before. The generation of new inventive designs is not possible.
- Closure of the possible solution space: Due to the rule-concept, the limits of the solution space are clearly visible since the rules function as pointer to each of the encoded solutions. On the other hand, rules may be used to code restrictions of the solution space like used in R1/XCON in the case of associations between two or more sub-components or in the extrusion profile example regarding the suppression state of the cut.
- Degree-of-Exploration: In principle, rules are able to address each of the variants within the solution space when used as decision tree. Since all possible variants are addressed, there are no degrees of freedom inside the solution space. This may be complicated in parametric design because every parameter value range has to be expressed incrementally which results in a big hierarchy of rules. The implementation of a decision table is a more compact formulation and has the same functionality. In terms of software engineering this concept corresponds to the select-case structure. On the other hand, when used as reasoning mechanism the exploration of the solution space is clearly structured but more flexible than in most of today’s common web-configurators. Like in R1/XCON and MYCIN, rules may be used for coding control knowledge that extends the simple procedural approach.
- Connectivity of multiple solution spaces: Basically, the interaction of solution spaces is not effected by use of a rule-based system. As the same at the size of the solution space, the interaction is documented but not widened or reduced. The system behavior is clear at all times since it is fully described by the rules. Nevertheless, as mentioned before, when rule-bases grow, the maintainability of the system declines. This is due to the fact, that every newly introduced rule has to be checked for consistency against the whole existing rule-base.

CONCLUSION

In the present article, the effects of rule-based systems on product complexity were discussed. Therefore, the Hannover House of complexity was introduced as a general framework for complexity management. For assessment, four measures have been presented that describe the possible design solution space for e.g. mass customization offers.

As noted before, the rule concept is one of the earliest implementations of knowledge-based systems and knowledge-based-engineering incorporated in an expert system. Today, rule-based concepts can be found in configuration systems, design support systems or in variant design automation. The fact, that rules are used as knowledge representation of heuristics and explicit design knowledge contributes to this.

By nature, product complexity can be reduced using rule-based systems since a solution space is formally described so that all possible solutions are known and in most cases the decisions of the reasoning mechanism are clearly understandable.

Nevertheless, a rule-base is nothing else than a pure description of an existing solution space. Regardless of being created manually or automatically, the rule-base has to address every feasibly design either through a consistency check or a decision tree. Creative design is out of their focus.

There exist a number of contributions which discuss the automatic generation of rule-bases for knowledge-based analysis systems, e.g. the assessment of a customer's credit ranking. A possible research question is how to transfer these fundamentals to KBE and synthesis systems respectively.

On the other hand, other KBE-mechanisms like constraint-based reasoning or case-based reasoning allow a different kind of formulation of solution spaces. These mechanisms of course have different effects on product complexity. Our present research targets on recommendations which KBE mechanism or product configuration approach is most useful for different types of business models and different types of design task.

Note: This paper is based on the paper presented at The 7th International Conference on Mass Customization and Personalization in Central Europe – MCP–CE 2016 – Mass Customization and Open Innovation, organized in Novi Sad, SERBIA, September 21-23, 2016.

References

[1.] C. Bliss, Management von Komplexität. Berlin, Heidelberg, Germany: Springer, 2000.

[2.] M. Marti, Complexity Management: Optimizing Product Architecture of Industrial Products. PhD-Thesis, Graduate School of Business Administration, Economics, Law and Social Sciences (HSG), University of St. Gallen, Switzerland, 2007

[3.] W. J. C. Verhagen, P. Bermell-Garcia, R. E. C. van Dijk, R. Curran, "A critical review of Knowledge-Based Engineering: An identification of research challenges" in Advanced Engineering Informatics, vol. 26, no. 1, pp. 5-15, 2012

[4.] G. Frizelle, The Management of Complexity in Manufacturing. London, United Kingdom: Business Intelligence Ltd., 1998

[5.] P. C. Gembarski, R. Lachmayer, "A Business typological Framework for the Management of Product Complexity" in Proceedings of the 8th World Conference on Mass Customization, Personalization, and Co-Creation (MCPC 2015), Montreal, Canada, October 20th-22nd 2015, 2015, pp. 1-13.

[6.] M. Hirz, Integrated Computer-aided design in automotive development. Berlin, Heidelberg, Germany: Springer, 2013

[7.] W. R. Ashby, An Introduction into Cybernetics. London, United Kingdom: Chapman & Hall, 1961

[8.] H. Bandte, Komplexität in Organisationen, Berlin, Heidelberg, Germany: Springer, 2007

[9.] M. Ghoffrani, Entwicklung und Einführung eines flexiblen Softwaresystems zur Konfigurierung virtueller Produkte, Aachen, Germany: Shaker, 2008

[10.] T. Jania, Änderungsmanagement auf Basis eines integrierten Prozess- und Produktdatenmodells mit dem Ziel einer durchgängigen Komplexitätsbewertung. PhD-Thesis, Institute of Product Development, University of Paderborn, Germany, 2005

[11.] G. Schuh, U. Schwenk, Produktkomplexität managen. Munich, Germany: Hanser, 2001

[12.] M. Gießmann, Komplexitätsmanagement in der Logistik. Cologne, Germany: Eul, 2010

[13.] C. Bliss, Management von Komplexität. Berlin, Heidelberg, Germany: Springer, 2000.

[14.] G. Schreiber, "Knowledge Engineering" in Handbook of knowledge representation, vol. 1, Elsevier, 2008

[15.] C. B. Chapman, M. Pinfold, "The application of a knowledge based engineering approach to the rapid design and analysis of an automotive structure" in Advances in Engineering Software, vol. 32, no. 12, pp. 903-912, 2001

[16.] W. Mettrey, "An Assessment of Tools for Building Large Knowledge-Based Systems" in Artificial Intelligence, vol. 8, no. 4, pp. 81-89, 1987

[17.] M. Cederfeldt, Planning Design Automation, PhD-Thesis, Product and Production Development, Chalmers University of Technology Göteborg, Sweden, 2007

[18.] J. McDermott, "R1: A Rule-Based Configurer of Computer Systems" in Artificial Intelligence, vol. 19, no. 1, pp. 39-88, 1982

[19.] W. J. Clancey, "The Epistemology of a Rule-Based Expert System – a Framework for Explanation" in Artificial Intelligence, vol. 20, no. 1, pp. 215-251, 1983

[20.] K. Laurischkat, Product-Service-Systems, IT-gestützte Generierung und Modellierung von PSS-Dienstleistungsanteilen. Aachen: Shaker, 2012

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

[22.] C. Xuewen, Z. Siyu, C. Jun, R. Xueyu, "Research of knowledge-based hammer forging design support system." in The International Journal of Advanced Manufacturing Technology vol. 27., no. 1, pp 25-32, 2005

[23.] R. Hunter Alarcón, J. Ríos Chueco, J. M. Pérez García, A. Vizán Idiope, "Fixtrure knowledge model development and implementation based on a functional design approach" in Robotics and Computer-Integrated Manufacturing, vo. 26, pp. 56-66, 2010

ANNALS of Faculty Engineering Hunedoara
– International Journal of Engineering
copyright © UNIVERSITY POLITEHNICA TIMISOARA,
FACULTY OF ENGINEERING HUNEDOARA,
5, REVOLUTIEI, 331128, HUNEDOARA, ROMANIA
http://annals.fih.upt.ro