Available online at www.sciencedirect.com

ScienceDirect

Procedia CIRP 21 (2014) 479 – 484

www.elsevier.com/locate/procedia

24th CIRP Design Conference

Application and validation of the matrix-based product description in a case study by using the software Loomeo

Thomas Luft[a]*, Niklas Ewringmann[a], Sandro Wartzack[a]

[a]*Friedrich-Alexander-University Erlangen-Nuremberg, Chair of Engineering Design, Martensstraße 9, 91058 Erlangen*

* Corresponding author. Tel.: +49-9131-85-23215; fax: +49-9131-85-23223. *E-mail address:* luft@mfk.uni-erlangen.de

---

Abstract

The main task of product developers is to meet the ever increasing demands on quality, cost and functionality of technical products by setting the appropriate characteristics in order to realize the product properties desired by the customer. The overall objective of the authors is to develop a suitable procedure model for property-based product development, which guides the developers purposefully through the development process. An essential part of this procedure model is the matrix-based product description, which is in the focus of this paper. The matrix-based product description is basically a so-called multi domain matrix. By using this matrix-based product description, dependencies between requirements, behaviors of the product and the product's properties and characteristics can be mapped and analyzed systematically during the product development process. Thereby, for example, the effects of characteristic changes on product properties become visible and traceable and also differential requirements are set into relationship with properties and characteristics of the product.

However, it has been shown that the manual filling of the matrix-based product description is very time-consuming. Hence, a computer-aided support is indispensable. For this reason, the software Loomeo is used in this paper to investigate its potential for IT support for the matrix-based product description. The interaction of dependencies between characteristics, properties and the resulting behavior is demonstrated in this paper through the matrix-based product description using the example of a front wheel suspension of a car. Thus, the strengths and weaknesses as well as opportunities and risks of Loomeo regarding the suitability of the IT-based matrix-based product description can be determined. The findings from both the application as well as from the evaluation of the software are the basis for the improvement and extension of the matrix-based product description.

Selection and peer-review under responsibility of the International Scientific Committee of "24th CIRP Design Conference" in the person of the Conference Chairs Giovanni Moroni and Tullio Tolio

Keywords: Design synthesis and optimisation; Design methodology; Product-oriented process management; Matrix-based product description; Design Structure Matrix; Complexity management; Case study.

Nomenclature

AS	Active structure
B	Behavior
C	Characteristic
CAM	Cambridge Advanced Modeller
CP	Component
DMM	Domain Mapping Matrix
DSM	Design Structure Matrix
F	Function
FS	Functional structure
MBPD	Matrix-based product description
MDM	Multi Domain Matrix
OS	Overall system
P	Property
PD	Product developer
PDP	Product development process
REQ	Requirement
S	Structure
SS	Subsystem

Selection and peer-review under responsibility of the International Scientific Committee of "24th CIRP Design Conference" in the person of the Conference Chairs Giovanni Moroni and Tullio Tolio
doi:10.1016/j.procir.2014.02.056

1. Introduction

The increasing individualization due to customer-specific requirements as well as the enormous complexity of modern products leads to steadily increasing challenges in the field of product development. While product and development cycles are becoming shorter, product developers (PD) have to meet the ever increasing demands on quality, cost and functionality of products. Hence, developers have to realize certain product properties by setting the appropriate characteristics [1] in order to fulfill the differentiated requirements of customers.

1.1. Problem statement and motivation

Due to these major challenges in product development and since the design problem is a multi-objective constraint satisfaction and optimization problem [2], PDs must be able to develop products that sufficiently meet customer needs efficiently and effectively. The behavior of the product caused by its properties under consideration of specific usage and environmental conditions is the most important criterion for the measurement of the fulfillment of customers' desires and requirements. Following the perspective according to Weber, appropriate characteristics have to be defined during the product development process (PDP) in order to achieve this required property profile of the product that arises from a variety of complex cause-effect chains resulting from a set of relations [1]. So, large and complex networks between characteristics (= causes) and resulting properties (= effects) quickly occur [3].

However, even minor modifications of (one single) characteristic(s) of one component can lead to a vast variety of changes concerning the property profile of the product. If the required property profile is not achieved by the product to be developed, PD cope in practice usually in the context of reworks (= iterations) only with the effects and do not eliminate their causes. The main reason for this is that developers are not able to identify and, after that, to retrace these causes fast and reliably in most cases because of the many different and complex dependencies between the defined characteristics and the resulting properties [4].

Consequently, taking appropriate action alternatives is not possible because consequences cannot be defined completely nor assessed adequately [5]. Hence, the effect of wrong decisions and unnecessary iterations increases in particular more complex structures.

1.2. Objectives

The overall objective is to develop an approach to improve the product-oriented process management. For this purpose, an advanced procedure model for property-based product development which is based on a matrix-based product description (MBPD) was proposed by Krehmer [5] and further developed by Luft [3, 4]. This procedure model guides developers purposefully through the development process and additionally supports the detection of the actual product maturity level and assists in the execution of necessary iterations [5].

The interaction of dependencies between characteristics, properties and the resulting behavior is demonstrated in this paper through the MBPD using the example of a front wheel suspension of a car. By applying this approach in this case study, the authors currently perform a first validation regarding the applicability, usefulness and effort of the MBPD. It became obvious that setting up a MBPD in an industrial case necessitates a computer-aided support in order to reduce the expenditure of time. That is why the software Loomeo is used in this contribution to investigate its suitability for the computer-aided MBPD.

2. State of the art and related work

After all existing design methodological approaches, processes and procedure models were described, analyzed and thereafter evaluated in detail regarding the aforementioned objectives in [3, 5], the authors were able to deduce the need for action for a novel procedure model for property-based product development. Following a brief explanation of the most important terms in section 2.1, the MBPD, which is the core element of the advanced procedure model, is explained in section 2.2.

2.1. Definitions of terms

The terms characteristics and properties have an inconsistent meaning in the literature on design methodology [1, 6, 7] and are used in different contexts. The understanding of the terms "characteristic" and "property" in this work is based largely on the definitions of Weber [1]. Consequently, the properties of a product cannot be determined directly but are the result of various characteristics which are determined directly by developers. The resulting property profile defines the product's behavior, which in the end is relevant for customers. Thereby, properties can be either quantitatively (e.g. cost, stiffness, weight) or qualitatively measurable (e.g. aesthetics, manufacturability, environmental friendliness). Following this understanding of the terms, it can be differentiated between intensive and extensive properties [8]. For instance, material characteristics belong to the intensive properties, which are the result of the selection of physic-chemical characteristics (e.g. choice of material). The extensive (or actually realized) properties of a component (e.g. stiffness, weight) arise from the combination of the intensive properties and the geometrical characteristics (e.g. length, width, height, shape). The (structural) dependencies between the (extensive) properties of the components are defined by the determination of structural characteristics (e.g. distance, angle). This leads to the properties of individual product modules and, in a further step, to the properties of the entire product [5] (see fig. 1).

So, the characteristics are the direct "setscrews" of PDs for determining the product's property profile. The behavior of the product is obtained as a result of the realized product properties by taking into account the actual usage (e.g. force by the user) and environmental conditions (e.g. road surface, climatic circumstances) [5].

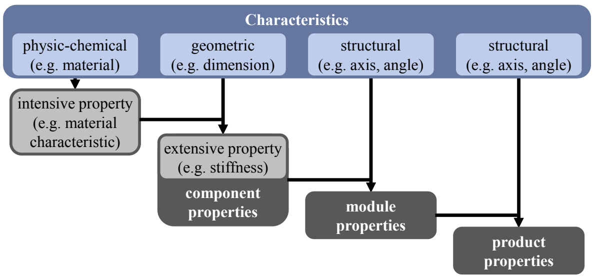


Fig. 1. Relations between characteristics and properties according to [4, 5]

2.2. The matrix-based product description

The advanced procedure model for property-based product development was proposed by Krehmer [5] and was further developed by Luft [3, 4] and consists basically of three main parts “procedure model”, “micro-cycles” and “matrix-based product description”. The procedure model is divided into several process steps and each one is assigned to one of the four perspectives behavior (B), properties (P), structure (S) and function (F) of the product. The purpose of the procedure model is to support PDs in the execution of the process steps. Each of these steps is structured by suitable and specific micro-cycles. This information is entered systematically into the MBPD, which is explained in the following. Detailed explanations of the procedure model and the micro-cycles can be taken from [3, 4, 5].

As part of the synthesis and starting from the requirements (REQ), the developers are guided step by step from the overall system (OS) level (e.g. racing car) to the subsystem (SS) level (e.g. chassis, power train) down to the component (CP) level of the product. The result of the procedural model with its associated micro-cycles is a completely filled MBPD. A simplified and schematic overview of such a MBPD is shown in the following figure (fig. 2).

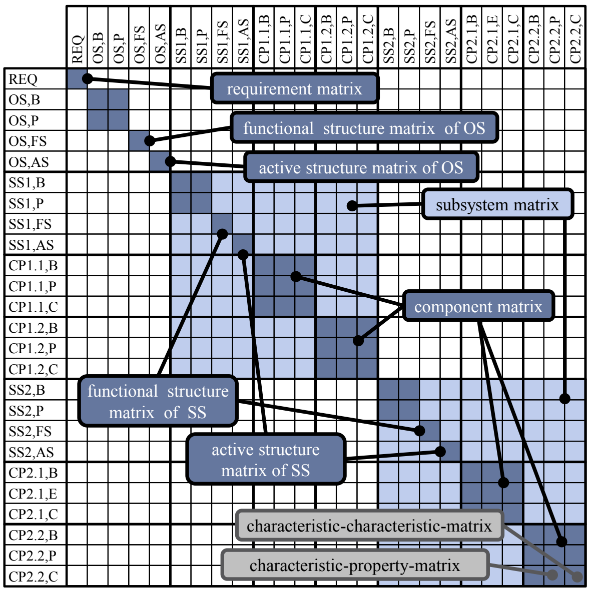
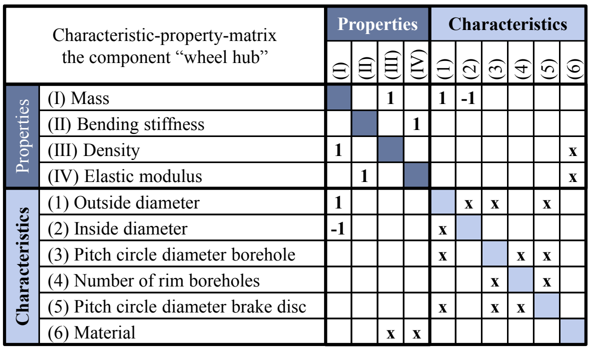


Fig. 2. Simplified and schematic overview of the MBPD according to [5]

The MBPD is basically a so-called multi domain matrix (MDM) and is composed of a variety of design structure matrices (DSM) which describe relationships between two elements of the same domain (characteristic-characteristic-matrix), and domain mapping matrices (DMM) that describe relationships between two elements of different domains (e.g. characteristic-property-matrices). By using this MBPD, dependencies between REQ, the behaviors (B) of the product, the product’s properties (P), the functional structure (FS) and the active structure (AS) as well as the characteristics (C) can be mapped and analyzed systematically during the PDP.

For instance, PDs have to create characteristic-property-matrices for all components because properties can be realized only through the determination of characteristics. Subsequently, all required characteristics (e.g. radius, material) are defined for the individual components (e.g. wheel hub) in characteristic-characteristic-matrices. Thereby, the effects of characteristic changes on product properties become visible and traceable as well as the numerous REQ are set into relationship with the properties and characteristics of the product at the OS-level, the SS-level and CP-level. The interaction of dependencies between characteristics, properties and the resulting behaviors is shown in this paper through the MBPD by using the example of a front wheel suspension of a racing car. A simplified characteristic-property-matrix for the component wheel hub is shown as an example in the figure below (fig. 3).



Characteristic-property-matrix the component “wheel hub”		**Properties**				**Characteristics**					
		(I)	(II)	(III)	(IV)	(1)	(2)	(3)	(4)	(5)	(6)
Properties	(I) Mass	■		**1**		**1**	**-1**				
	(II) Bending stiffness		■		**1**						
	(III) Density	**1**		■							**x**
	(IV) Elastic modulus		**1**		■						**x**
Characteristics	(1) Outside diameter	**1**				▨	**x**	**x**		**x**	
	(2) Inside diameter	**-1**				**x**	▨				
	(3) Pitch circle diameter borehole					**x**		▨	**x**	**x**	
	(4) Number of rim boreholes							**x**	▨	**x**	
	(5) Pitch circle diameter brake disc					**x**		**x**	**x**	▨	
	(6) Material			**x**	**x**						▨

Fig. 3. Characteristic-property-matrix of a wheel hub (simplified extract)

By using, for example, a characteristic-property-matrix, it can be analyzed which unintended effects on properties have proposed modifications to certain characteristics. Here, different types of dependencies or interactions can be distinguished from each other. Thus, the influences or dependencies between two elements are marked with an “x”. In this case, it is not possible to indicate the direction of the dependency that is necessary to distinguish whether the element A is dependent on element B or if element B is dependent on element A. In contrast to “x”, the direction of the dependency can be specified by a plus sign “+1” or a minus sign “-1”. For instance, a negative directed dependency (“-1”) exits when the value of a dimensional characteristic (e.g. material) increases and, as a result of this, the value of a property (e.g. density) decreases. As a result, positive and negative correlations between elements (e.g. characteristics and properties can be modeled by using the MBPD.

3. Application of the software Loomeo

Systems in industrial practice include far more elements than the simple examples for illustration purposes in the previous chapter. Since setting up of a complex system with all its elements and dependencies is very time-consuming [9], a computer-aided support is essential.

3.1. Suitable tools for setting up and analyzing a MBPD

There are a variety of software tools for modelling and analyzing interactions, dependencies and flows in complex systems (e.g. products, processes, organizations). Some of them have been developed in particular for supporting researchers and practitioners in working with DSMs, DMMs and MDMs. For these tasks, there is a comprehensive list of tools on the DSM website at www.dsmweb.org. Here, a distinction is made between research (e.g. Cambridge Advanced Modeller (CAM), Matlab and Excel Macros) and commercial (e.g. Acclaro DFSS, Lattix and Loomeo) tools.

These software tools have been evaluated by the authors with respect to certain criteria which can be determined before an application. This rough evaluation has revealed that Loomeo enables a holistic modeling of DSMs, DMMs and MDMs, which is essential for setting up a MBPD. Moreover, this tool also provides other important functionalities for the MBPD (e.g. a function for deriving indirect dependencies). Therefore, the authors have decided to apply and validate first of all Loomeo (despite the relatively high purchase price).

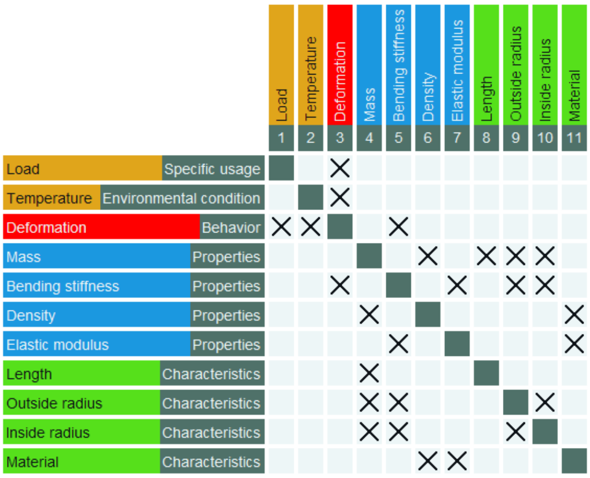
3.2. The software Loomeo

A first evaluation of the MBPD by using the above mentioned case study has already been done in [4]. The authors currently perform further validations regarding the applicability, usefulness and effort of the MBPD. However, it has already been shown that the manual collection of all the characteristics, properties, behavioral aspects and further information as well as the numerous dependencies and relationships between them is very time-consuming. Hence, when setting up a MBPD in an industrial environment, a computer-aided support for reducing the time required is indispensable. For this reason, the software Loomeo from Teseon GmbH is used in this paper to investigate its potential for IT support for the MBPD. Hence, the focus in this paper is the validation of the application by using the Loomeo.

This software is focused on structural complexity management and particularly used for documentation, visualization and analysis of complex dependencies. By using matrices, force-directed graphs and diagram elements of systems and their dependencies can be represented and manipulated. Loomeo also allows developers to create a MDM-structure because this software is based on and supports the terminology of the DSM-methodology.

3.3. Setting up a MBPD by using Loomeo

A suspension arm, which is mounted transversely to the driving direction and is connected articulated with the chassis and the wheel carrier, is used in this paper as an application example. The decisive condition of use is therefore the load due to forces. All other usage conditions will be neglected. As an environmental condition, only the (ambient) temperature is considered in this application example. These two usage and environmental conditions are highlighted in orange in the following MBPD in which no distinction between certain types of dependencies is considered (fig. 4).



		Load 1	Temperature 2	Deformation 3	Mass 4	Bending stiffness 5	Density 6	Elastic modulus 7	Length 8	Outside radius 9	Inside radius 10	Material 11
Load	Specific usage	■		×								
Temperature	Environmental condition		■	×								
Deformation	Behavior	×	×	■		×						
Mass	Properties				■		×		×	×	×	
Bending stiffness	Properties			×		■		×		×	×	
Density	Properties				×		■					×
Elastic modulus	Properties					×		■				×
Length	Characteristics				×				■			
Outside radius	Characteristics				×	×				■	×	
Inside radius	Characteristics				×	×				×	■	
Material	Characteristics						×	×				■

Fig. 4. MBPD of the wishbone by using the software Loomeo (an extract)

The deformation of the wishbone is the main CP behavior (highlighted in red). Due to the load, the wishbone is deformed elastically or plastically under excessive stress. The deformation contributes to the transmission behavior of the wheel suspension. The considered behavior of the product, the deformation, is resulting among others from the properties “bending stiffness” and “axial stiffness” (not shown). Some properties of the wishbone like “mass”, “density” and “Young's modulus” are shown in blue. For instance, a dependency between the properties “bending stiffness” and “Young's modulus” can be modeled with the MBPD (fig. 5).

By selecting a material and three geometrical characteristics, PDs can directly determine the characteristics of the (simplified) suspension arm. The tube (as base body) is sufficiently described by the “length” and the “outside radius” and “inside radius”. Here, the outside radius is, for example, dependent on the inside radius because the reduction of the outer radius requires also the reduction of the inner radius in order to retain a constant wall thickness of the tube.

3.4. Analyzing a MBPD by using Loomeo

In addition to setting up the MBPD, Loomeo can also be used to support the graphical representation, analysis, optimization and improvement of the MBPD. Therefore, not only algorithms but also different forms of representation are suitable to identify and measure interactions [10].

The representation in matrix form is not appropriate in case of larger MBPD for achieving a clear and graphical illustration of dependencies. For this, Loomeo provides, alternatively to the matrix, a graph representation in which nodes and edges are represented in graphical form (fig. 5). This visualized structure or network of the MBPD is much more intuitive to capture by the PD and therefore more suitable to get a “picture” of the whole MBPD with all its dependencies and relations.

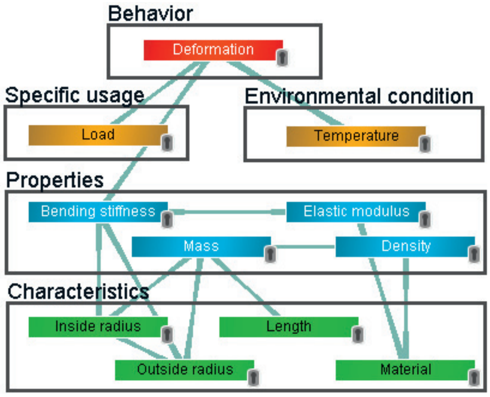


Fig. 5. Analysis of the wishbone with the graph representation in Loomeo

By means of the generally netlike graph representation of a MBPD (e.g. for a component), the challenges of the PD become apparent. Characteristics (e.g. inside radius) do not only influence one or more properties (e.g. bending stiffness), but are also dependent on other characteristics (e.g. outside radius). The characteristics depending on this characteristic are, in turn, affecting other properties. Using this form of the matrix representation, PDs can easily identify characteristics which need to be varied to achieve specific product properties according to the REQ regarding the product’s behavior. Thereby, it is important to note that one single modification of a characteristic may affect a large number of further elements. These effects may not always be problematic and partially evaluated intuitively by PDs, but they require attention and may have far-reaching consequences on SS- and OS-level.

In addition, two mathematical algorithms for analyzing the MBPD were applied in this application example. It has been shown that a MBPD created with Loomeo can be analyzed not only by using clustering algorithms but also by partitioning algorithms. Clustering is generally used to group elements in such a way that elements in the same cluster are more similar (e.g. have more dependencies) to each other than to those in other clusters. Such groups could provide an indication of setting a modular product design. By the application of partitioning algorithms onto a matrix, proposals can be made to an appropriate processing sequence of the PDP. It has to be mentioned that the use of clustering and partitioning has much more potential at OS- and SS-level than at CP-level of the MBPD.

4. Validation of the software Loomeo

4.1. Criteria for the validation of Loomeo

Some evaluation criteria are described in the following for an initial assessment of the applicability of Loomeo. Two main phases can be roughly distinguished when working with the MBPD. In the system modeling phase, relevant elements (e.g. characteristics) of the system and known dependencies between them are identified and entered into the MBPD. The analysis and improvement of MBPD takes place in the system analysis phase [7]. Due to the occurrence of changes between modeling and analysis during the PDP, the elements of a MBPD (e.g. characteristics, properties) are constantly modified by PDs. The third group of criteria is derived from general requirements for software. The three groups with an extract of the validation criteria used are listed in Fig. 6. In the following sections, the first use of Loomeo for the computer-aided MBPD will be evaluated according to these criteria.

System modeling	Handling and clarity (e.g. clear representation of large data sets)
	Types of matrices and their linkage (e.g. DSM, DMM, MDM)
	CAx-interfaces (e.g. use of data from CAD-programs)
System analysis	Analysis of nodes and edges (e.g. various computer-based calculations)
	Support of different mathematical algorithms (e.g. clustering, banding)
	Handling of dependencies (e.g. determining, representation)
General criteria	Costs (e.g. software or license fees, support costs)
	Range of functions (e.g. quantity and quality of functions)
	User interface (e.g. graphical user interface, intuitive usability)

Fig. 6. Criteria for the validation of Loomeo (an excerpt)

4.2. Validation of Loomeo regarding system modeling

Basis for assessing the suitability of the software in terms of system modeling should be the experience from its application in the case study. The terminology in Loomeo is consistent and is based strictly on the theoretical foundations of the DSM methodology. The software basically allows the disclosure of information concerning nodes and edges. Consequently, a detailed description of the interactions and dependencies is possible. However, the PD is not actively supported. The design and handling of the program, however, is intuitive and therefore the system modeling is largely self-explanatory. A further benefit of Loomeo is the simple and consistent transition between matrix and graph representation. In both forms it is possible to modify the MBPD and its structure and elements. While Loomeo offers good representations for small matrices with few elements, the clarity decreases rapidly with an increasing number of elements and matrices. A disadvantage is also the limited scalability and that there is no possibility to group domains.

An advantage of Loomeo is that all necessary types of matrices as well as types of dependencies for the MBPD are fully supported and made available for PDs. The entire MDM, however, can be modeled only as a matrix of domains (e.g. characteristics, properties) and not as a matrix of all elements. Loomeo allows users to model different and multiple types of dependencies. A benefit is that the weighting of dependencies is possible. In addition, different elements of DSMs, for example, can be linked together by means of further data sets. Since Loomeo also includes the automated linking of DSMs and DMMs, the modeling of the MBPD is greatly simplified.

A disadvantage of the software is that exchanging data is difficult because of missing interfaces between Loomeo and CAx-systems [11]. Although there is no direct interface between Loomeo and Excel, there are possibilities to edit DSMs and DMMs via intermediate steps in Excel.

4.3. Validation of Loomeo regarding system analysis

In the context of system analysis, the second group of criteria, Loomeo offers comprehensive possibilities for analyzing nodes and edges. For instance, the calculation of the active- and passive sum as well as of the criticality is callable directly from the matrix representation and can be integrated into the MBPD. The active and passive values of the elements of the MBPD can be used to identify the degree to which elements affect other elements in the MBPD (active impact) and are themselves affected by other elements (passive impact). Furthermore, this software supports the application of various mathematical operations such as "creation of weighted sums" or "determination of the maximum value". These results can be visualized with appropriate diagrams.

When analyzing the MBPD by mathematical algorithms, the user has to abstain from banding algorithms. Clustering and partitioning, however, are available in Loomeo. These two types of algorithms for optimizing the structure of the MBPD are less suited for analysis of single components (e.g. characteristic-characteristic-matrices) but are rather designed for analysis at SS- und OS-level.

Strengths of Loomeo are that dependencies can be changed via an easy to use form as well as the simple derivation and clear representation of (indirect) dependencies which are explained in detail in [4] and [12]. It is advantageous that the MBPD can be represented both as a matrix and as a graph. The identification, analysis and handling of interactions and dependencies are supported by these types of visualization. In addition, it is possible to consider and to analyze positive and negative correlations between various elements (fig. 3).

4.4. Validation of Loomeo regarding general criteria

With regard to some selected general criteria, it can be stated that Loomeo is relatively expensive in relation to other software, like research tools, that support the computer-aided MBPD with similar functions. The range of functions of Loomeo is sufficient, since the necessary matrices for the MBPD can be used and analyzed. As pictograms are integrated in the user interface, available functions of Loomeo can be recognized easily by PDs. Unfortunately, the menu also consists of icons, which still have faulty functionality. However, the user interface can be described as user-friendly.

5. Summary and outlook

Overall, it can be summarized that Loomeo is appropriate for the computer-aided MBPD. Due to the fundamental support of the generic approach of DSM-methodology almost any MBPD can be created and analyzed by using Loomeo. However, this software shows weaknesses for large MBPD in terms of a clear representation and is very expansive. Thus, most of the strengths and weaknesses as well as opportunities and risks of Loomeo regarding the suitability of the IT-based MBPD are determined. The findings from both the application as well as from the evaluation of the software are the basis for the improvement and extension of the computer-aided MBPD. Since this proposed approach enables a comprehensible MBPD, the interdisciplinary collaboration is enhanced as all PDs involved can participate in the creation of this MBPD.

Future research work will deal not only with the further validation of the MBPD by using Loomeo but also with the validation of CAM. Then, Loomeo and CAM are compared with each other through a comprehensive evaluation. Since DSM has been used to model many different types of systems [13], the proposed MBPD will be also linked in future work to the steps of PDP, the development organization and the so-called knowledge and information objects (cf. [11, 12]).

Acknowledgements

The authors thank the German Research Foundation (DFG) for supporting this research project with the title "Product-oriented process management – iterations management based on a property-based product maturity".

References

[1] Weber C. CPM/PDD - An extended theoretical approach to model-ling products and product development processes. In: Bley H, Jansen H, Krause F, Shpitalni M, editors. Proceedings of 2nd German-Israeli Symposium on Advances in Methods and Systems for Development of Products and Processes. Stuttgart: Fraunhofer; 2005, p. 159-179.

[2] Qureshi AJ, Eisenbart B, Dantan JY, Blessing, L. Design Automation with the Characteristics Properties Model and Property Driven Design for Redesign. In: Abramovici M, Stark R,editors. Proceedings of the 23rd CIRP Design Conference. Springer. 2013. p. 563-572.

[3] Luft T, Krehmer H, Wartzack S. An advanced procedure model for property-based product development. In: Lindemann U, Venkataraman S, Kim YS, Ion W, Malqvist Y, editors. Proceedings of the 19th International Conference on Engineering Design. Seoul: Design Society; 2013. p. 305-314.

[4] Luft T, Wartzack S. Die matrixbasierte Produktbeschreibung als Bestandteil des Vorgehensmodells in der eigenschaftsbasierten Produktentwicklung. In: Spath D, Bertsche B, Binz H, editors. Stuttgarter Symposium für Produktentwicklung. Stuttgart: Fraunhofer; 2013.

[5] Krehmer H. Vorgehensmodell zum Iterations- und Produktreifegrad-management in der eigenschaftsbasierten Produktentwicklung. Erlangen: VDI-Verlag, Fortschrittsberichte VDI, Reihe 1, Band 416; 2012.

[6] Hubka V. Theorie technischer Systeme. Grundlagen einer wissen-schaftlichen Konstruktionslehre. 2th ed. Berlin: Springer; 1984.

[7] Ponn J. Lindemann U. Konzeptentwicklung und Gestaltung technischer Produkte. 2th ed. Berlin: Springer; 2011.

[8] Roozenburg N. Defining synthesis: on the senses and the logic of design synthesis. In: Chakrabarti A, editor. Engineering Design Synthesis - Understanding, Approaches and Tools. London: Springer; 2002. p. 3-18.

[9] Lindemann U, Maurer M, Braun T. Structural complexity management. An approach for the field of product design. Berlin: Springer; 2009.

[10] Al-Zaher A, ElMaraghy W, Pasek ZJ, ElMaraghy H. Design of Recon-figurable Automotive Framing System. In: Abramovici M, Stark R,editors. Proceedings of the 23rd CIRP Design Conference. Springer. 2013. p. 253-262.

[11] Amigo CR, Iritani DR, Rozenfeld H, Ometto A. Product Development Process Modeling: State of the Art and Classification. In: Abramovici M, Stark R,editors. Proceedings of the 23rd CIRP Design Conference. Springer. 2013. p. 169-179.

[12] Luft T, Bochmann J, Wartzack S. Enhancing the flow of information in the PLM by using numerical DSMs – an industrial case study. Proceedings of the IFIP WG5.1 – 10th International Conference on Product Lifecycle Management. Nantes: Springer, 2013.

[13] Eppinger S, Browning T. Design Structure Matrix Methods and Appli-cations. Cambridge: MIT Press. 2012.