INTERNATIONAL DESIGN CONFERENCE - DESIGN 2018
https://doi.org/10.21278/idc.2018.0298

SEMI-AUTOMATIZED ASSESSMENT OF REQUIREMENT INTERRELATIONS

I. Graessler, P. Scholle, J. Hentze and C. Oleff

Abstract

Changing requirements have a broad impact on product development processes. In this paper, a novel approach towards structuring requirements is proposed. Based on a requirements list, interrelations of requirements are assessed semi-automatically by a rule basis. Here, generic interrelations funded on either physical fundamentals or working principles are recorded. By this approach, requirements structure matrices are derived semi-automatically. Combined with selecting critical requirements based on structured criterions, iterations due to changing requirements will be reduced.

Keywords: requirements management, structuring, design structure matrix (DSM), risk management, additive manufacturing

1. Introduction

Volatile economic environment and decreasing length of product lifecycles demand for shortening the time to market. Simultaneously, the complexity of products and systems is increasing. In new product development (NPD), requirements, and their changes can exert a wide impact on product development (Song et al., 2017).

In both, conventional engineering methodology (such as VDI 2221 (1993), Pahl/Beitz (2007)), and interdisciplinary approaches (such as VDI 2206 (VDI, 2004) or Systems Engineering (SE Handbook Working Group, 2011)), the central role of requirements for NPD is emphasized. The chronology of every NPD project is unique due to a variety of customer- and project induced targets and restrictions (Albers, 2010). Project delays and iterations are often caused by changing requirements as a result of dynamic underlying needs and restrictions for the new product (Wynn et al., 2007). The success of complex NPD projects, highly depends on requirement management (Brauns, 2016). Due to changing or incomplete requirements, nearly 24% of all projects are delayed (Pohl, 2010). Requirements cannot be considered independent: By either physical fundamentals or working principles, requirements are interrelated (Scholle et al., 2015). Requirement interrelations are the basis for assessing the impact of a change of single requirements on (multiple) other requirements. Regarding physical fundamentals, the impact of a geometric change on the weight of a system can be mentioned - with an increase in size, the weight will be affected. If working principles such "leverage" occur within a system, the geometry of the lever will automatically influence the realizable force. Once such influences are documented, potential future developments of requirements can be anticipated. Thereby the impact of a changing requirement onto other requirements of a complex technical system can be assessed. Furthermore, appropriate reaction strategies can be derived, to minimize potential negative impacts (Gräßler et al., 2017).

In this paper, a novel approach for a semi-automatized assessment of requirement interrelations is presented. This approach is based on a pre-defined set of rules which enable the user/development team to assess interrelations between requirements (semi-)automatically. The overall benefit of this approach is a distinct reduction of assessment effort for requirements interrelations. Additionally, risks caused by changing requirements can be anticipated. Thereby, iterations in NPD can be reduced.

The research for this paper is structured according to the Design Research Methodology (DRM) proposed by Blessing and Chakrabarti (2009). In Section 2, the relevant state of the art of requirements engineering, requirements structuring and semi-automatized assessment of requirement interrelations is described (Descriptive Study I); followed by the description of a novel approach for a semi-automatized assessment of requirement interrelations (Section 3, Prescriptive Study). The approach is tested using the practical example of a wheel carrier for a race car in Section 4. Finally, an outlook on future research is given.

2. State of the art

The results of the first step of DRM - the Descriptive Study I - is presented in this section Requirements engineering describes the process of elicitating and structuring requirements (Pohl, 2010). Management/Treatment and documentation of requirements is subsumed under the aspect of requirements management (Pohl, 2010). Originating from software engineering, various approaches towards requirements engineering have been developed (Rupp, 2007; Pohl, 2010). Depending on the focus of NPD, tailored approaches for product-service-systems (Berkovich, 2012) or electrical engineering (Allmann, 2008) were presented. In other methodical approaches, such as property driven development (PDD), the central role of requirements as a control parameter for NPD is emphasized (Weber and Deubel, 2003).

Requirements can be structured in multiple ways. The central role of the requirements list is outlined by Pahl et al. (2007). Here, requirements are documented according to a pre-defined structure (Figure 1). Requirements may be sorted according to their relevance (demands and wishes).

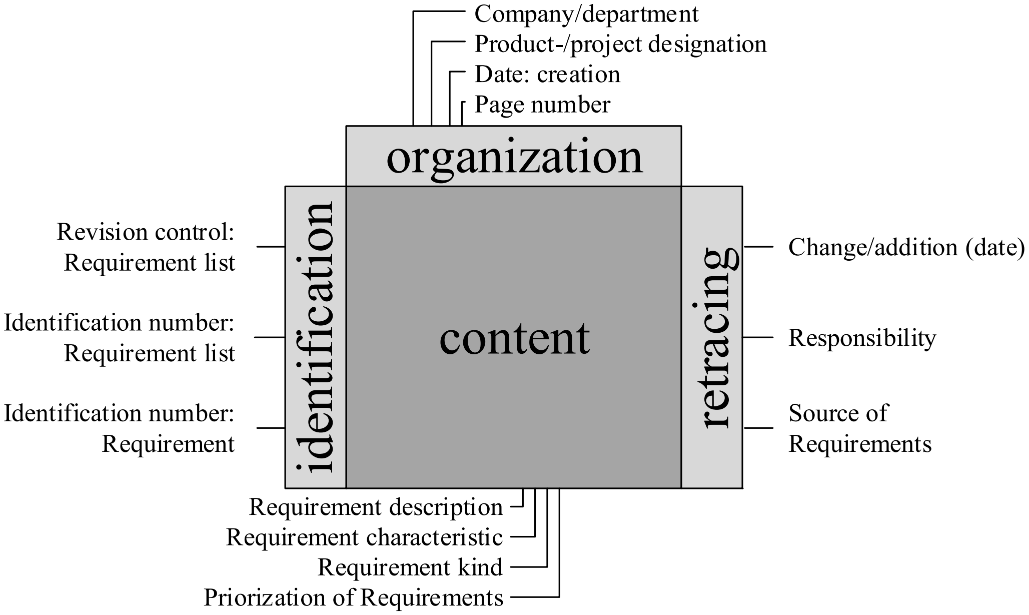

Company/department
Product-/project designation
Date: creation
Page number
organization
Revision control: Requirement list
Identification number: Requirement list
Identification number: Requirement
identification
content
retracing
Change/addition (date)
Responsibility
Source of Requirements
Requirement description
Requirement characteristic
Requirement kind
Priorization of Requirements

Figure 1. Structure of a requirements list (Pahl et al., 2007)

The "Checklist for setting up requirement lists" is proposed as a method for setting up requirements lists and ensuring their comprehensiveness by Pahl et al. (2007). The checklist containing 156 elements in 21 subcategories is shown in Figure 2.

Material

- Initial and final product
- Chemical properties
- Physical properties
- Auxiliary materials
- Prescribed material
- Flow and transport of mate

Energy

- Performance
- Loss
- Efficiency
- State variables
- Pressure
- Temperature
- Heating
- Cooling
- Supply
- Storage
- Capacity
- Conversion

Signal

- In- and outputs
- Display
- Equipment
- Control equipment
- Form

Geometry

- Dimensions
- Diameter
- Space
- Number
- Arrangement
- Connection
- Extension

Mechanical properties

- Weight
- Load
- Forces
- Dynamic
- Static
- Friction
- Thermal stress
- Stability
- Resistance
- Deformation
- Stiffness
- Kinematic
- Type and direction of motion
- Acceleration
- Velocity
- Kinetic
- Suspension
- Resonances

Electric / Electronic

- Voltage
- Current
- Power fluctuations
- Fuse
- Shield
- Filtering
- EMV
- Connection
- Wiring
- Isolation
- Creeping Distance
- Plug
- Modularization
- Functional groups
- SMD-Parts
- Availability of parts
- Accessibility
- Exchangeability

Software

- Integration
- Interfaces
- Updates
- Hardware
- Testing
- Emergency operation

Safety

- Direct safety
- Indirect safety
- Indicative safety
- Operational safety
- Occupational safety
- Environmental safety
- Potential threat
- Risk
- Risk assessment

Ergonomics

- Human-machine-interaction
- Display and interfaces:
- Control
- Method of control
- Clear layout
- Luminance
- Anthropometric measure
- Operating forces
- Tactile coding
- Feel

Industrial Design

- Importance
- Aesthetics
- Indication features
- Symbolic features
- Recognition
- Colour scheme
- Sinus-Milieu

Procurement

- Make-or-buy strategy
- A-Suppliers
- Local content
- Pre-defined assemblies
- Operational / strategic procurement
- Data exchange

Production

- Factory limitations
- Maximum possible dimensions
- Preferred production methods
- Means of production
- Achievable tolerances and quality

Assembly

- Special regulations
- Installation
- Assembly
- Sitting
- Foundations
- Tools
- Auxiliary material
- Safety instructions

Maintenance

- Servicing intervals
- Inspection
- Replacement vs. restoration
- Cleaning
- Lubrication
- Operation site

Recycling

- Reuse
- Disposal
- Final disposal
- Hazardous material
- Critical material for recycling
- Accessibly
- Disassembly

Transport

- Limitations due to lifting gear
- Clearance
- Means of transport (height and weight)
- Despatch
- Delivery time

Operation

- Quietness
- Wear
- Special uses
- Operation site
- Humidity
- Services

Control

- Possibility of testing and measuring
- Special regulations and standards

Planning

- Maximum production cost
- Tool costs
- Investments
- Amortisation
- Deadline
- Delivery date
- Project plan
- Specific know-how

Sustainability

- LCA
- Efficiency
- System costs

Market

- Competitors
- Customer segments
- Customer behaviour and needs
- Competitive standard
- Sales

Figure 2. Checklist for setting up a requirements list (Feldhusen and Grote, 2012)

A combination of requirements elicitation and their relation to the environment of a product is presented by various authors (Humpert, 1995; Jung, 2006). Humpert and Jung trace the relations between the environment and the requirements. In their model-based approach, requirements are derived from an environment model, which is related to a requirement model. The completeness of the requirements list is improved by this approach.

A comprehensive approach for modelling and structuring requirements is presented by Stechert (2010). Requirements are elicitated, processed and provided by a requirement model. The model is based on Systems Modeling Language (SysML) (Stechert, 2010). This model is used as a basis for a combination of modelling requirements and product variants targeting complexity management (Huth et al., 2017). In Model-Based Requirements Engineering, interrelations between requirements are derived from the system model (Holt et al., 2015). A model-based multi-disciplinary modelling technique for requirements management is proposed by Hackenberg et al. (2014).

Despite assessing the relations between requirements and system elements or functions from a model-based perspective, interrelations between requirements may also be assessed. Based on Browning's approach towards Design Structure Matrices (DSM), a matrix-based approach for modelling requirement interrelations is presented by Eben et al. (2010). At first, relevant domains (stakeholders) are collected. Interrelations between different domains are assessed in a Multiple Domain Matrix (MDM) (Eben et al., 2010). Hereafter, requirements are elicitated from selected domains. Interrelations between requirements are assessed in a requirement DSM. Requirements are then prioritized according to their interrelations and the originating domain.

A matrix-based approach towards prioritizing requirements was presented by Eben and Lindemann (2010). Requirements are taken as nodes of a graph. Interrelations between requirements represent dependencies. By applying graph-theoretical approaches such as active or passive sum, clustering or identification of paths, the structure of the requirement graph can be analysed. (Eben and Lindemann, 2010)

According to contemporary state-of-research in software engineering, requirements and their interrelations are analysed to evaluate the propagation of requirement changes (Knethen and Grund, 2003; Goknil et al., 2014; Zhang et al., 2014). Various dependency types are identified (Pohl, 1996; Dahlstedt and Persson, 2005; Zhang et al., 2014). Dependent on interrelations, requirements are prioritized (Shao et al., 2016). Additionally, the stability of a software release is assessed on basis of requirements stability indices (Christoper and Chandra, 2012). Inconsistencies between requirements are identified by analysing their interrelations (Goknil et al., 2011; Ali et al., 2013; Escalona et al., 2013). Approaches in the field of software engineering have to be adapted for the development of interdisciplinary cyber-physical systems.

An approach towards integrating, structuring, analysing and maintaining requirements for the development of cyber-physical systems based on the requirement dependency model by Pohl (1996) was proposed by Gräßler and Hentze. Here, requirement interrelations are assessed in a Requirements Structure Matrix (RSM) (Gräßler and Hentze, 2017). A RSM - alike a DSM - is a square matrix with both, row and column labels representing single requirements. As in a DSM, the off-diagonal mark is an indicator for the strength of the interrelation of one requirement on another (Browning, 2001). Setting up such an RSM requires effort from an interdisciplinary team of development experts (Gräßler and Hentze, 2017). A tool for a semi-automatized assessment of requirement interrelation is proposed by Knethen and Grund (2003). Thereby, interrelations for assessing traceability in software requirements engineering are automatically detected on the basis of a software representation of requirements in RE tools such as IBM Rationale DOORS or Requisite Pro.

3. Method for semi-automatized assessment of requirement interrelations

As outlined in Section 2, the assessment of requirement interrelations is challenged by various perspectives: Especially in small and medium-sized enterprises (SME), the usage of RE tools is uncommon. To reduce the necessary effort to set-up a RSM, approaches towards an automatized detection of requirement interrelations are necessary. An automatized detection of requirement interrelations improves the RSM approach in multiple ways:

- reduction of effort required for setup of RSM
- formalization of underlying (implicit) expert knowledge
- shift of workload towards value contributing tasks of NPD

The approach for a semi-automatized assessment of requirement interrelations is presented in the following sections. The method is explained in Section 3.1. The underlying set of rules for semi-automatized assessment is addressed in Section 3.2.

3.1. Method

For NPD in small and medium-sized enterprises (SMEs), the checklist proposed by Pahl et al. (2007) is a common method for the elicitation of requirements. In order to integrate the proposed method into the processes of SMEs, the method was developed in addition to the checklist. The method is presented in Figure 3.

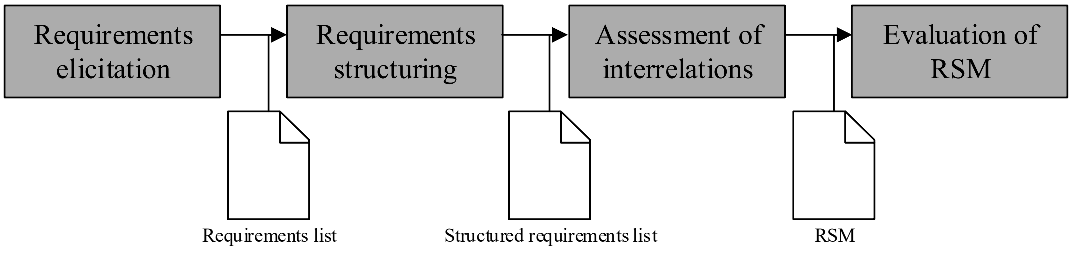


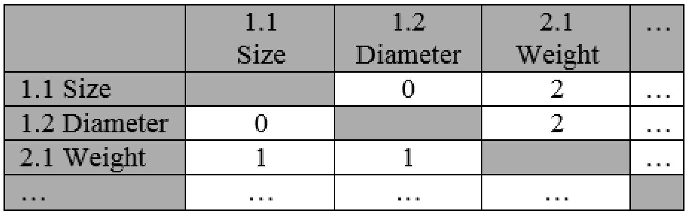
Figure 3. Method for semi-automatized assessment of requirement interrelations

In the first step, requirements are elicitated according to the elements in the "Checklist for setting up requirements lists". Requirements are then added to a requirements list and structured accordingly.

Table 1. Structured requirements list (acc. to Pahl et al., 2007)

#	Requirements	D/W	Responsible
1	Geometry		
1.1	Length		
1.2	Diameter		
…	…	…	…
2	Forces		
2.1	Weight		
…	…	…	…

Based on the structure presented above , requirements are semi-automatically assessed by applying the set of rules described in Section 3.2. Outcome of this step is a RSM. Similar to the square character of a DSM, the RSM represents interrelations of single requirements. For this approach, the RSM is considered as a weighted matrix. An element value of "zero" represents no interrelation, while "one" and "two" represent medium and strong interrelation of the requirements. The RSM is directed. The influence of one requirement on another can be different from the influence vice versa. The RSM for the requirements list above is shown in Figure 4.



	1.1 Size	1.2 Diameter	2.1 Weight	…
1.1 Size		0	2	…
1.2 Diameter	0		2	…
2.1 Weight	1	1		…
…	…	…	…	

Figure 4. Requirements structure matrix (RSM)

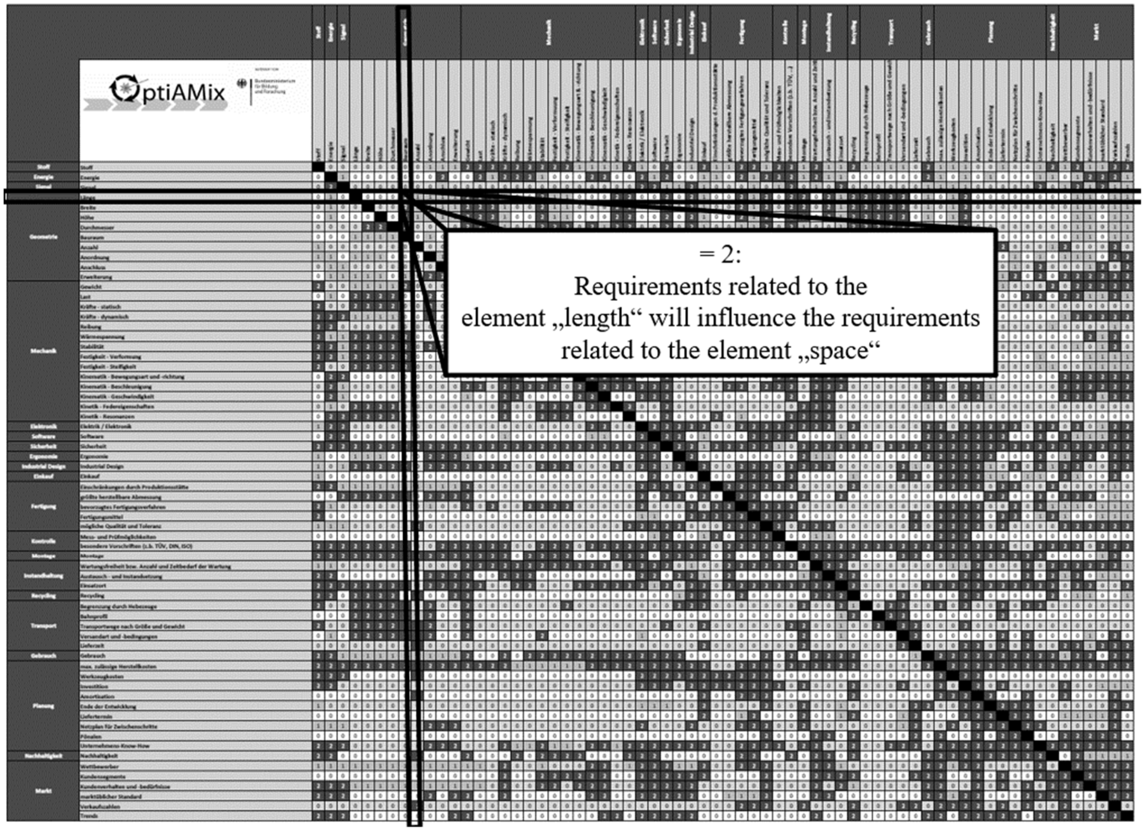
Evaluating the RSM is the next step of the underlying process. By analysing various criteria such as activity or passivity of a requirement, determining the potential risk of a change of a single requirement and the resulting impact on other requirements is assessed. Criteria for structural analysis are based on work done by Lindemann and Eben (Lindemann et al., 2009; Eben and Lindemann, 2010). The criteria selected are presented in Table 2 below.

Table 2. Structural criterions for the evaluation of the RSM (acc. to Eben and Lindemann, 2010)

Criterion	Explanation	Meaning
Active sum	Sum of outgoing relations	A requirement with a high active sum has an impact on other requirements.
Passive sum	Sum of ingoing relations	A requirement with a high passive sum is influenced by other requirements.
Criticality	Multiplication of active sum and passive sum (Lindemann et al., 2009)	A requirement with a high criticality affects and is affected by a large number of other requirements. Because of its high importance in the system it should be given a high priority (Eben et al., 2010)
Reachable node	Number of nodes reached directly or via possible paths (Lindemann et al., 2009)	Influence of requirement on others within the whole model, impact of its change (Eben and Lindemann, 2010)

3.2. Rule set for semi-automatized assessment

The set-up of a RSM is usually done manually by an expert team. To reduce this effort, a set of rules for a semi-automatized assessment of requirement interrelations has been developed. The set is funded on the "Checklist for setting up requirements lists", which is used for structuring the requirement list in previous steps of the process described in Section 3.1. To ensure a broad applicability of the rule set in various projects, interrelations between the different elements within the checklist were assessed. Interrelations were identified by literature study targeting physical fundamentals and working principles. In order to reduce the complexity, not all of the 21 categories of the checklist from the eight German edition of Pahl & Beitz (see Feldhusen and Grote, 2012) were considered in detail. For instance, the main category "Software" with the elements "integration", "interfaces", "updates", "hardware", "testing" and "emergency operation" was not detailed. The relationship with other elements in the checklist was just assessed for the category "Software" but not for each element within. Hereby, the complexity of the rule set could be reduced for a first version. Overall, 65 elements in 21 subcategories were assessed. An overall of $65^2 = 4225$ potential interrelations were identified. The interrelations were evaluated on a scale from "zero" to "two", "zero" represents no interrelation while an entry with "two" as a value means strong interrelation. The rule set is shown in Figure 5 below.



= 2:
Requirements related to the
element „length“ will influence the requirements
related to the element „space“

Figure 5. Rule set for semi-automatized assessment of interrelations in the RSM

Usage of the standardized rule set allows a semi-automatic transfer of structured requirement lists into RSMs. Dependening on the checklist assigned to single requirements, interrelations can be exerted from the rule set and inserted into the RSM. The RSM derived by applying the rule set is then presented to the developer. By this, generic interrelations can be adapted to the specific use case.

4. Validation

In this section, the approach is applied to a case study from a development project carried out by students. The product is a wheel carrier of a race car (Figure 6) designed by members of the university's racing team "UPBracing".

Figure 6. Wheel carrier of student's race car

The requirements list contained 25 requirements for the additively manufactured wheel carrier. After assigning elements from the checklist to the requirements, the RSM was derived by applying the rule set presented in Section 3.2. The RSM for the wheel carrier is a 25x25 square matrix. The graph related to the RSM is shown below.

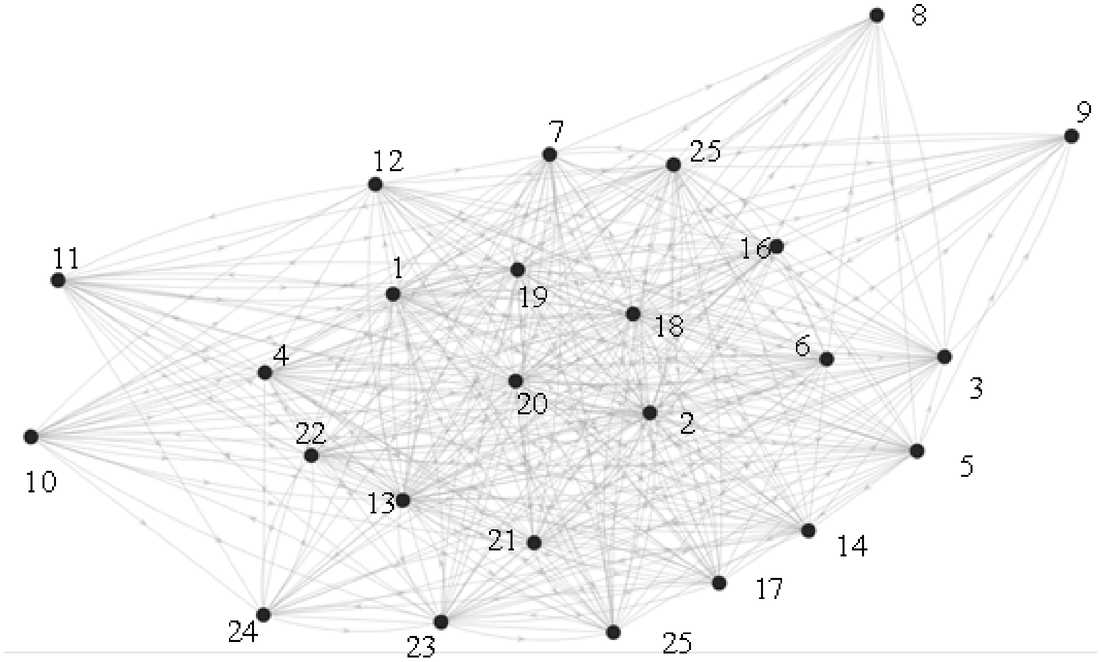


Figure 7. Graphical representation of the RSM for the wheel carrier

Requirements related to the elements "Preferred production technology" (Requirements 20, 21, 22) and "Material" (Requirements 1 and 2) are characterized by the highest active sum. Passive sum for requirements related to the element "Stability" (Requirement 13) is highest.

Changes in requirements with highest active sum affect many other requirements. In contrast, the requirements related to the element "Stability" (Requirement 13) has many ingoing relations. Thereby, it is likely to be affected by changing requirements during the development project. Implementation and validation activities for this requirement should - if possible - be postponed to a later point of the project to reduce iterations. Elsewise, the requirement (or the affecting requirements) should be closely monitored throughout the project to reduce the risk.

5. Summary and outlook

In this paper, an approach towards a semi-automatized assessment of requirement interrelations was presented. The approach is based on a requirement list structured according to the "Checklist for setting up requirements lists" by Pahl (2007). After elicitation of requirements and structuring them in a list according to the before mentioned checklist, a det of rules is applied. Thereby, interrelations between elements in the checklist are identified and assessed. Furthermore a Requirements Structure Matrix (RSMs) can be derived semi-automatically based on the underlying set of rules. In a final step. The RSM can be evaluated according to various structural criteria. Thereby, requirements with a high impact on other requirements can be identified.

Benefit of the method's application is a reduction of iterations in development projects induced by requirement changes. The effort required to set up the RSM is minimized due to the semi-automatic character of the approach. Relevant expert knowledge is formalized by using the rule set. Workload is shifted from supporting to mainly operational tasks in new product development. For the developer, interrelations between requirements are illustrated in an efficient way. Hereby, the capability to determine the effect of changing requirements on other requirements and the development process itself is increased.

Future work is focused on empirical validation in a two-step approach: First, students were asked to set up a requirements list for a specific product and provide a RSM. Second, results from the semi-automatically derived RSMs will be compared to the RSMs provided. Thereby, the quality of the rule set can be assessed and improved. In a second step, the rule set will be applied to real design cases of a varying complexity: Starting with components of limited complexity, the rule base will be applied to complex design projects with a high number of requirements later on. Increasing the rule set to additional elements in the checklist and to implement further sources of knowledge such as working principle-induced requirement relations is in focus of future work. Relevant fields for development of mechatronic systems such as software-related categories tackled by VDI Technical Committee 4.10 "Interdisciplinary Product Creation" will be the focus of future work. Currently, the approach is implemented in a prototypical Visual-Basic application. Next steps include the transcription into a software tool linked to other requirement tools such as IBM rationale doors or other tools used by project partners for implementation.

The application of the approach is not only limited to derivation of RSMs and evaluation of requirements. Resulting RSMs can be used as input to scenario-based requirements. In addition, potential future developments of requirements and the resulting impact on others can be assessed (Gräßler et al., 2017). Combining both approaches will be part of further research.

Acknowledgements

This research was enabled by the funding of the German Ministry for Education and Research (BMBF) in the context of the project OptiAMix ("Mehrzieloptimierte und durchgängig automatisierte Bauteilentwicklung für additive Fertigungsverfahren im Produktentstehungsprozess") within the programme "Additive Fertigung - Individualisierte Produkte, komplexe Massenprodukte, innovative Materialien (ProMat_3D)".

References

Albers, A. (2010), “Five Hypotheses about Engineering Processes and their Consequences”, *Proceedings of the TMCE 2010, Ancona, I, April 12 - 16, 2010.*

Ali, R., Dalpiaz, F. and Giorgini, P. (2013), “Reasoning with contextual requirements. Detecting inconsistency and conflicts”, *Information and Software Technology*, Vol. 55 No. 1, pp. 35–57, https://doi.org/10.1016/j.infsof.2012.06.013.

Allmann, C. (2008), *Situations- und szenariobasiertes Anforderungsmanagement in der automotive Elektronikentwicklung*, Audi Dissertationsreihe, Cuviller Verlag, Göttingen.

Berkovich, M. (2012), Requirements Engineering für IT-gestützte Product Service Systems, PhD thesis, Techhnische Universität München, München, 2012.

Blessing, L.T.M. and Chakrabarti, A. (2009), *DRM, a Design Research Methodology*, 1st ed., Springer, London, https://doi.org/10.1007/978-1-84882-587-1

Brauns, C. (2016), Requirements Engineering und Management in der wehrtechnischen Beschaffung. Empirische Bestandsaufnahme, Konzeptentwicklung, Evaluation, PhD thesis, Helmut-Schmidt-Universität / Universität der Bundeswehr Hamburg.

Browning, T.R. (2001), “Applying the design structure matrix to system decomposition and integration problems: a review and new directions”, *IEEE Transactions on Engineering Management*, Vol. 48 No. 3, pp. 292–306. https://doi.org/10.1109/17.946528

Christoper, F.X. and Chandra, E. (2012), “Prediction of software requirements stability based on complexity point meassurement using multi-criteria fuzzy approach”, *International Journal of Software Engineering & Applications (IJSEA)*, Vol. 3 No. 6, pp. 101–115. https://doi.org/10.5121/ijsea.2012.3608

Dahlstedt, Å.G. and Persson, A. (2005), “Requirements Interdependencies: State of the Art and Future Challenges”, In: Aurum, A. and Wohlin, C. (Eds.), *Engineering and managing software requirements*, Springer, Berlin, pp. 95–116. https://doi.org/10.1007/3-540-28244-0_5

Eben, K.G.M. and Lindemann, U. (2010), “Structural Analysis of Requirements. Interpretation of structural criterions”, *12th International Dependency and Structure Modelling Conference DSM '10*, Vol. 1, pp. 249–261.

Eben, K.G.M., Daniilis, C. and Lindemann, U. (2010), “Interrelating and Prioritising Requirements on Multiple Hierachy Levels”, *Proceedings of the Design 2010 / 11th International Design Conference, Dubrovnik, Croatia, May 17 - 20, 2010*, The Design Society, Glasgow, pp. 1055–1064.

Escalona, M.J., Urbieta, M., Rossi, G., Garcia-Garcia, J.A. and Luna, E.R. (2013), “Detecting Web requirements conflicts and inconsistencies under a model-based perspective”, *Journal of Systems and Software*, Vol. 86 No. 12, pp. 3024–3038. https://doi.org/10.1016/j.jss.2013.05.045

Feldhusen, J. and Grote, K.-H. (2012), *Pahl/Beitz Konstruktionslehre: Methoden und Anwendung erfolgreicher Produktentwicklung*, Springer, Berlin.

Goknil, A., Kurtev, I., van den Berg, K. and Veldhuis, J.-W. (2011), “Semantics of trace relations in requirements models for consistency checking and inferencing”, *Software & Systems Modeling*, Vol. 10 No. 1, pp. 31–54. https://doi.org/10.1007/s10270-009-0142-3

Goknil, A., Kurtev, I., van den Berg, Klaas and Spijkerman, W. (2014), “Change impact analysis for requirements: A metamodeling approach”, *Information and Software Technology*, Vol. 56 No. 8, pp. 950–972. https://doi.org/10.1016/j.infsof.2014.03.002

Gräßler, I. and Hentze, J. (2017), “Structuring and Describing Requirements in a Flexible Mesh for Development of Smart Interdisciplinary Systems”, *8th ECCOMAS Thematic Conference on Smart Structures and Materials (SMART17), Madrid, Spain, June 5 - 8, 2017,* pp. 1622-1631.

Gräßler, I., Scholle, P. and Pottebaum, J. (2017), “Integrated process and data model for applying scenario-technique in requirements engineering”, *Proceedings of the 20th International Conference on Engineering Design (ICED 15), Vancouver, Canada, August 21 - 25, 2017*, The Design Society, Glasgow, pp. 261–270.

Hackenberg, G., Richter, C. and Zäh, M.F. (2014), “A Multi-disciplinary Modeling Technique for Requirements Management in Mechatronic Systems Engineering”, *Procedia Technology*, Vol. 15, pp. 5–16. https://doi.org/10.1016/j.protcy.2014.09.029

Holt, J., Perry, S., Payne, R., Bryans, J., Hallerstede, S. and Hansen, F.O. (2015), “A Model-Based Approach for Requirements Engineering for Systems of Systems”, *IEEE Systems Journal*, Vol. 9 No. 1, pp. 252–262. https://doi.org/10.1109/JSYST.2014.2312051

Humpert, A. (1995), *Methodische Anforderungsverarbeitung auf Basis eines objektorientierten Anforderungsmodells*, HNI-Verlagsschriftenreihe, Bd. 9, Heinz-Nixdorf-Institut, Paderborn.

Huth, T., Inkermann, D. and Vietor, T. (2017), “Ein Ansatz für eine integrierte, modellbasierte Anforderungs- und Variantenmodellierung”, *Wissenschaftsforum Intelligente Technische Systeme (WInTeSys): 11. und 12. Mai 2017 Heinz Nixdorf MuseumsForum, Paderborn, Paderborn, May 11 - 12, 2017,* Heinz-Nixdorf-Institut, Paderborn, pp. 169–182.

Jung, C. (2006), Anforderungsklärung in interdisziplinärer Entwicklungsumgebung, PhD thesis, Technische Universität München.

Knethen, A. von and Grund, M. (2003), “QuaTrace: a tool environment for (semi-) automatic impact analysis based on traces”, *International Conference on Software Maintenance, 2003. ICSM 2003. Proceedings, Amsterdam, Netherlands, September 22 - 26, 2003*, pp. 246–255. https://doi.org/10.1109/ICSM.2003.1235427

Lindemann, U., Maurer, M. and Braun, T. (2009), *Structural complexity management: An approach for the field of product design*, Springer, Berlin. https://doi.org/10.1007/978-3-540-87889-6

Pahl, G., Beitz, W., Feldhusen, J. and Grote, K.-H. (2007), *Engineering design: A systematic approach*, 3rd ed., Springer, Berlin.

Pohl, K. (1996), *Process-centered requirements engineering*, Advanced software development series, Vol. 5, Research Studies Press, Taunton.

Pohl, K. (2010), *Requirements engineering: Fundamentals, principles, and techniques*, Springer, New York. https://doi.org/10.1007/978-3-642-12578-2

Rupp, C. (2007), *Requirements-Engineering und -Management: Professionelle, iterative Anforderungsanalyse für die Praxis*, 4th ed., Hanser, München.

Scholle, P., Song, Y.-W., Herzog, M., Bender, B. and Gräßler, I. (2015), “Methoden der Anforderungsstrukturierung zur Steuerung von Produktentwicklungsprozessen”, *Design for X: Beiträge zum*

26. DfX Symposium, Bd. 26, Herrsching, Deutschland, October 04 - 05, 2015, TuTech Innovation, Hamburg, pp. 121–132.

SE Handbook Working Group (INCOSE) (2011), *Systems Engineering Handbook - A Guide for System Life Cycle Processes And Activities*, San Diego.

Shao, F., Peng, R., Lai, H. and Wang, B. (2016), "DRank. A semi-automated requirements prioritization method based on preferences and dependencies", *Journal of Systems and Software*, Vol. 126, pp. 141-156. https://doi.org/10.1016/j.jss.2016.09.043

Song, Y.-W., Chahin, A., Scholle, P., Bender, B., Gräßler, I. and Paetzold, K. (2017), "Optimierung des Produktentwicklungsprozesses mittels Risikoanalyse vernetzter Anforderungen", *Design for X: Beiträge zum 28. DfX Symposium, Bd. 28, October, 2017,* TuTech Innovation, Hamburg, pp. 339–351.

Stechert, C. (2010), "Modellierung komplexer Anforderungen", PhD Thesis, Technische Universität Braunschweig.

VDI (1993), *VDI 2221 Systematic approach to the development and design of technical systems and products No. VDI 2221*, Verein Deutscher Ingenieure, Düsseldorf.

VDI (2004), *VDI 2206 Design methodology for mechatronic systems No. 2206*, Verein Deutscher Ingenieure, Düsseldorf.

Weber, C. and Deubel, T. (2003), "New theory-based concepts for PDM and PLM", *Proceedings of the 14th International Conference on Engineering Design (ICED 03), Stockholm, Sweden, August 19 - 21, 2003*, The Design Society, Glasgow.

Wynn, D.C., Eckert, C. and Clarkson, J. (2007), "Modelling Iteration in Engineering Design", *Proceedings of the 16th International Conference on Engineering Design (ICED 07), Paris, France, July 28 - 31, 2007*, The Design Society, Glasgow.

Zhang, H., Li, J., Zhu, L., Jeffery, R., Liu, Y. et al. (2014), "Investigating dependencies in software requirements for change propagation analysis", *Information and Software Technology*, Vol. 56 No. 1, pp. 40–53. https://doi.org/10.1016/j.infsof.2013.07.001

Philipp Scholle, M. Sc.
Heinz Nixdorf Institute - Paderborn University, Chair for Product Creation
Fuerstenallee 11, 33102 Paderborn, Germany
Email: philipp.scholle@hni.upb.de