Holistic change propagation and impact analysis in requirements management

Abstract
In developing complex technical systems, requirements are subject to continuous change. Systematic and holistic change impact analysis and proactive measures are required for reducing the number of requirement changes and their negative impact. There is no method to analyse the holistic impact of a requirement change in the context of developing complex technical systems. Holistic analysis requires to consider the local effects of requirement changes as well as effects from change propagation. To develop an approach for holistic change propagation and impact analysis, twelve performance goals are defined. Those are derived from a state of research analysis as well as an industry workshop. A three-step method is proposed. Firstly, requirement dependencies that cause change propagation are detected. Secondly, critical requirements are automatically identified based on a Page Rank algorithm. Thirdly, change impact of critical requirements is analysed based on a guideline. Validation proves that ten goals are fulfilled and two are partly fulfilled. The method addresses major shortcomings of preceding research and enables sound decision making for development engineers both before a change occurs and during decision process on a change request. This helps to reduce negative change impact in development projects and the risk of project failure.

Introduction
Requirements of radical innovation projects like developing complex technical systems are subject to continuous change. Each requirement change may cause additional effort. To reduce the number of requirement changes and negative impact, proactive measures carry great potential (Graessler, Oleff, and Scholle 2020). Systematic and holistic change impact analysis is required to utilize that potential.

Requirement changes are a key driver for project failure (The Standish Group 2017). Impact analysis of requirement changes for proactive risk management has few scientific contributions yet, but holds great potential to reduce negative impact (Hein, Voris, and Morkos 2018). Impact analysis means identifying potential consequences of a change, including local change extent and effects from change propagation. Areas of relevance are Engineering Change Management (ECM) (Hamraz, Caldwell, and Clarkson 2013) and Requirement Change Management (RCM) (Jayatilleke and Lai 2018). Specific tools in these areas are interdisciplinary but for system elements and with expert analysis (ECM), or highly automated but limited to software requirements with insufficient analysis of requirement network characteristics (RCM). Only few approaches include effects directly caused by the respective requirement change (local change effects), although this is a key driver (Graessler, Oleff, and Scholle 2020). Network analysis has shortcomings regarding differentiation of dependency types, higher order change propagation and ability to process large requirement sets.

There is no method to analyse the holistic impact of a requirement change in the context of developing innovative technical systems. Many approaches address propagation analysis, but lack the ability to process interdisciplinary requirements automatically, to differentiate types of requirement dependencies and to assess local change effects. Therefore, the following research questions are derived:

- RQ-1: "How can the impact of requirement changes in developing innovative complex technical systems be assessed holistically in early development stages?". Holistic analysis requires determining all possible change effects: local and propagation.
- RQ-1.1: "How can local effects of requirement changes be determined?"
- RQ-1.2: "How can requirement change propagation effects be determined?"

Methodology and Materials

The research methodology is derived from the Design Research Methodology (Blessing and Chakrabarti 2009) as part of a prescriptive study of a superordinate type five research project. A literature review is conducted to evaluate current solutions and research gaps and to elicit performance goals for holistic impact analysis. To address research gaps, approaches from gap specific areas of relevance are analysed (e.g. critical path analysis). Based on the findings, a method for holistic change impact analysis is developed and validated by two workshops with industry. Experts from a leading engineering service provider of automotive industry assess plausibility and applicability. The scientific approach is shown in Figure 1.

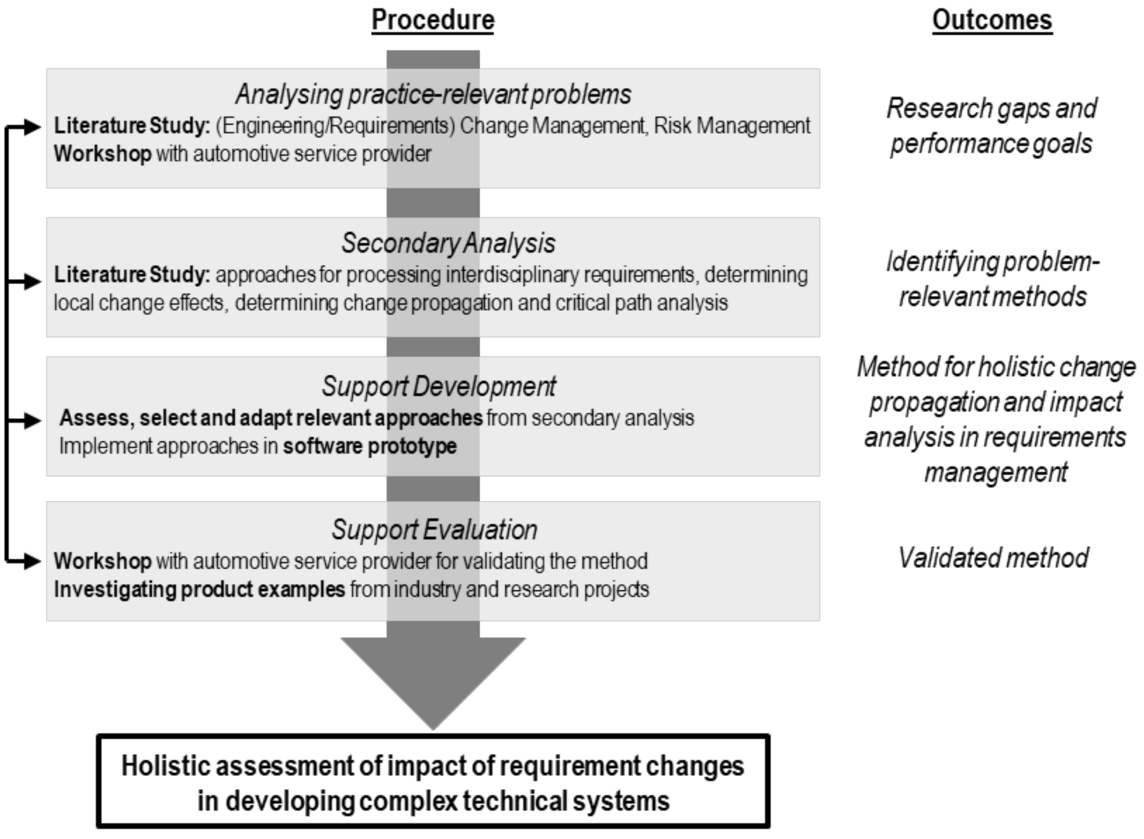

Figure 1: Research approach

28 relevant publications in ECM and 13 relevant publications in RCM were identified in the research repositories Google Scholar, IEEE Explore, Science Direct, Springer Link, and Web of Science. For evaluating the developed method, requirement sets of product examples from publicly funded research projects were investigated: two industry examples of automotive drive train sensor development (each about 1000 requirements). One further example was taken from a formula student racing team development project in an inverter casing (32 requirements with

change history) and an intelligent robotic arm (50 requirements). Moreover, two workshops with industry were conducted:

- First workshop for requirement elicitation and prioritization.
- Second workshop for validation of the method

State of Research

BOHNER & ARNOLD defined change impact analysis as "the activity of identifying the potential consequences, including side effects and ripple effects, of a change, or estimating what needs to be modified to accomplish a change before it has been made" (Bohner and Arnold 1996). This definition was used as a starting point to identify basic elements of change impact analysis that need to be investigated. It was gradually enriched by context specific findings (Graessler, Oleff, and Scholle 2020; Hein, Voris, and Morkos 2018; Clarkson, Simons, and Eckert 2004; Wickel 2017; Neumann 2017). This resulted in the understanding that impact analysis requires two consecutive steps: identification of change effects and analysis of the resulting impact:

- For **identification of change effects**, the change request (initial effect) as well as dependencies between requirements (consecutive effect) are investigated. Identification of change effects defines the areas of interest for impact analysis by indicating elements which potentially need to be changed. Identification of effects differs in terms of depth and width. Depth is characterized by the order of consecutive effects considered (e.g. direct dependencies vs. higher order dependencies). Width is characterized by the element types under investigation (e.g. different requirement types).
- After identification of effects, **impact analysis** aims to assign expected implications on project goals for each effect (e.g. quality, cost and time). This requires to predict decisions on how to implement the change (local effects and resulting impact) as well as ripple effects of those decisions (change propagation). The local change is influenced by the incoming change impulse, requirement characteristics (e.g. priority or tolerances) and development decisions (e.g. degree of change resolution). By adding up the local impact of the initial change and all subsequent changes, the collective impact can be assessed and compared to alternative solutions (e.g. different decisions on implementation or rejection of change).

The overall understanding of basic elements of change impact analysis is illustrated in Figure 2.

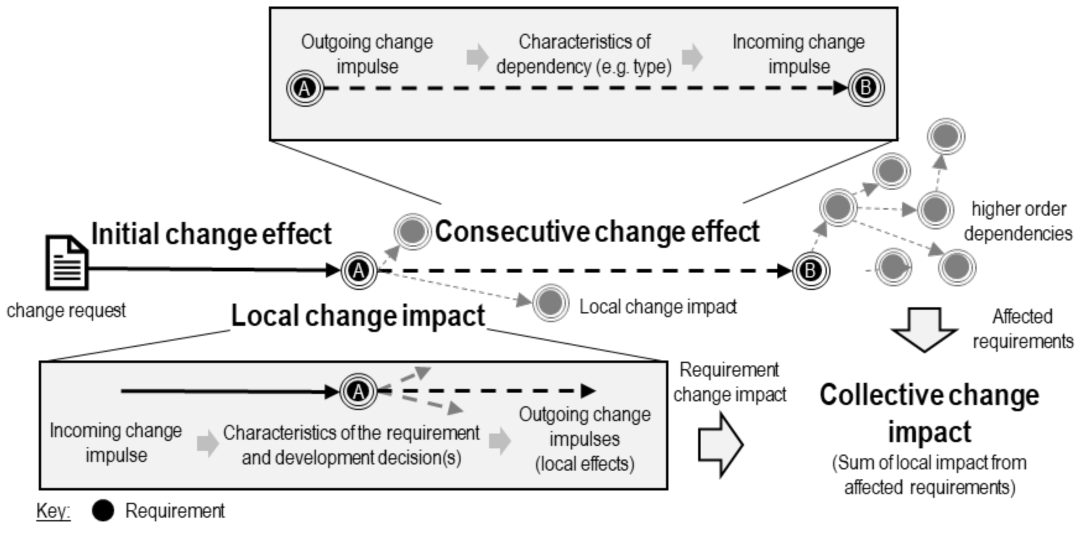

Figure 2: Basic elements of change impact analysis

Both steps strongly depend on the underlying **information base**. The information base can be expert knowledge as well as development data (e.g. requirement descriptions or requirement diagrams). It differs in scope (e.g. differentiation of dependency types), objectivity and processability (e.g. by algorithms). Availability and accuracy of the information base is limited in the early development stages and increases in the course of a development project. The lack of information in early development stages is especially distinctive for interdisciplinary products with various information types and areas of relevance as well as for projects with a high degree of novelty and uncertainty (radical innovations).

Existing methods for impact analysis can be evaluated and differentiated. A frequently cited method for impact analysis in ECM is the "**Change Prediction Method**" (CPM) according to CLARKSON ET AL. (Clarkson, Simons, and Eckert 2004). It enables risk assessment of change propagation for components based on expert evaluation. As a method, Design Structure Matrixes (DSM) are used for identification and characterization of direct dependencies and resulting change impact and probability. This requires little input data but in-depth system knowledge, which is not available in early development stages. Multiple extensions of CPM exist, mainly contributing to extend the scope of analysis regarding requirements, functions, or properties – e.g. (Lemmens et al. 2007; Koh, Caldwell, and Clarkson 2012; Hamraz et al. 2013). Those extensions still lack the ability to differentiate specific influence factors in the context of requirement changes. Examples are the differentiation of types of requirement dependencies for consecutive effects or the analysis of collective change impact. Additionally, extensions require structured input data, which is not available in the early development stages.

ECM approaches mentioned focus on consecutive effects from change propagation solely and lack the ability to assess local effects and collective change impact. The analysis of collective change impact is only supported by few approaches – e.g. (Gärtner et al. 2008; Browning and Eppinger 2002) – and requires extensive input-data to apply Monte Carlo simulation. These approaches are unfeasible for impact analysis of requirement changes, since that information is not available in industry in early development stages (Graessler, Oleff, and Scholle 2020; Hellenbrand, Helten, and Lindemann 2010).

A literature review on **RCM** (Jayatilleke and Lai 2018) shows similar tendencies towards stressing consecutive effects and propagation analysis but leaving local effects and collective impact unconsidered – e.g. (Eben and Lindemann 2010; Morkos 2012; Morkos, Mathieson, and Summers 2014). Approaches that do consider the impact rely on software specific information like use case maps or sizing of software code and thus cannot be applied for interdisciplinary requirement sets. The overview of existing approaches emphasizes three issues:

1. capability to analyse extensive interdisciplinary requirement sets of complex systems automatically and differentiate dependency types regarding propagation behavior,
2. consideration of local change effects as part of a holistic impact analysis and
3. preventive risk management for requirement changes in early development phases instead of reactive change treatment.

Performance Goals

28 requirements for the approach to be developed are defined on the basis of the state of research analysis (RA) as well as an industry workshop (IW). Following the approach of (Balzert 2009), requirements are prioritized as essential, conditionally necessary, or optional. Requirements are for example: "The approach should include company-specific knowledge" and "Requirement dependencies should be distinguished with respect to propagation behavior". A pairwise comparison of the requirements is performed to determine the priority. This is done exclusively

by the industry representatives to consider the importance of the requirements from a user perspective. Then, taking into account the user prioritization and the number and priority of the underlying requirements, twelve performance goals are derived. These are formed from a set of high-priority requirements. In order to consider different roles in the development, eight industry representatives from all task areas relevant to the application are involved in the workshop: head of department, project manager, requirements manager, process owner, product owner and three engineers. State of research points out performance goals related to the method and the available scope of information in early development stages. The industry workshop is used to especially integrate performance goals related to application of the method but also the input data format and output plausibility and comprehensibility.

Table 1: Performance goals for holistic change propagation and impact analysis

Category	**Performance Goal**	**Source**
Input	PG-1: Only natural language requirement specification and expert knowledge is needed	IW
Method	PG-2: Processability of complex requirement sets (high number of requirements from different disciplines)	IW, RA
	PG-3: Analysis of consecutive effects	RA
	PG-4: Analysis of local effects	RA
	PG-5: Differentiation of requirement dependency types	RA
	PG-6: Consideration of higher order requirement dependencies	RA
	PG-7: Analysis of collective impact	RA
Application	PG-8: Acceptable application effort	IW
	PG-9: Availability of required information and data	IW, RA
	PG-10: Availability of required software	IW
Result	PG-11: Plausible results	IW, RA
	PG-12: Comprehensible results	IW

Holistic change propagation and impact analysis

Applying the method for holistic change propagation and impact analysis, critical requirements regarding change propagation are identified and the change impacts of critical requirements are analysed. A requirement is classified as critical, if the change of this particular requirement is likely to induce multiple change impulses into the system or effects requirements with a high priority. Change behavior of requirements appears to be inseparable from developers' decisions and cannot be automated (Oehmen and Lindemann 2016). Therefore, only critical requirements regarding change propagation are assessed to minimize application effort. A Requirements Structure Matrix (RSM) is required as the data basis for the method. In this matrix, different dependency types – "requires", "is required by", "refines", "is refined by" and "none" – are mapped between requirements. These dependency types are transformed into weights corresponding to their degree of propagation (Goknil et al. 2014; Zhang et al. 2014). The dependency types “requires”, “is required by” and “is refined by” propagate inevitably and are therefore weighted with a value of one. Non-propagating dependency types – “none” and “refines” – are weighted with a value of zero. A RSM can be created using approaches described in a previously conducted literature study (Gräßler, Preuß, and Oleff 2020).

A modified Page Rank algorithm based on (Gräßler et al. 2019; Xing and Ghorbani 2004) is used to **identify critical requirements regarding change propagation**. The Page Rank is used as an indicator of the probability of change propagation of a requirement. For the application of the Page Rank algorithm, the RSM has to be analysed as a graph. The requirements represent nodes and the weighted dependencies represent the edges between nodes, respectively interrelations between requirements. Based on the graph, the sets E_i^{in} and E_i^{out} are defined as the set of inbound and outbound edges of a requirement r_i. A specific interdependency – e.g. “requires” – between requirements r_i and r_j is therefore set as v_{ij}. For the total number of *n* requirements and a damping factor d, the Requirement Page Rank $PR(r_i)$ for the requirement r_i is computed in equation (1). The Page Rank is an iterative method. Initial values have to be chosen. In this case, all nodes are assigned the same weight $\frac{1}{n}$ (Gräßler et al. 2019). The value of *d* is set to 0.85, as it proved to be a robust choice in this context (Gräßler et al. 2019).

$$PR(r_i) = \frac{1-d}{n} + d \sum_{r_j \in E_i^{in}} \left(\sum_{r_k \in E_j^{out}} \frac{PR(r_j)}{v_{kj}}\right) \tag{1}$$

Besides the effect of changing requirements on a specific requirement, the effect of a specific requirement on other requirements has to be considered in change propagation analysis. For this effect, the Requirement Active Rank $AR(r_i)$ of a requirement is calculated by interchanging the sets E_i^{in} and E_i^{out} in the equation above. The Active Rank $AR(r_i)$ calculates the likelihood a requirement r_i induces a change impulse within the system. In order to take into account not only the probability of propagation of requirement changes but also the impact, the priority of a requirement is additionally included in the calculation of the Active Rank. The values v_{ij} representing specific dependencies between requirements r_i and r_j are weighted with a factor $P(r_i)$ based on the priority of r_i. Optional requirements are weighted with a factor $P(r_i)$ of 0.5 and mandatory requirements with a factor $P(r_i)$ of one. This is referred to as the weighted Active Rank $AR_w(r_i)$ (see equation (2)). Requirements with a high weighted Active Rank $AR_w(r_i)$ are critical regarding change propagation. For classification as a critical requirement, an initial threshold of 0.8 is set. This threshold resulted from expert discussion in the second industry workshop, but requires further parameter studies to determine appropriateness in different application contexts.

$$AR_w(r_i) = \frac{1-d}{n} + d \sum_{r_j \in E_i^{out}} \left(\sum_{r_k \in E_j^{in}} \frac{AR_w(r_j)}{P(r_k) \cdot v_{kj}}\right) \tag{2}$$

Figure 3 shows an exemplary requirements network including weighted Active Rank. The diameter of a requirement node corresponds to the weighted Active Rank. Critical requirements are thus visualised with a particularly high diameter in the requirements network.

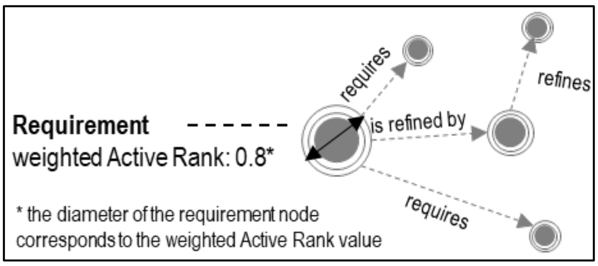

Figure 3: Requirements including weighted Active Rank

To minimize application effort in **analysing change impact**, only critical requirements are selected for expert analysis. To support the expert analysis and increase objectivity, a guideline is developed. A requirements engineer is confronted with the requirements network of critical requirements and the respective requirements descriptions of the network. The critical requirements are highlighted. Using a guideline, the requirements engineer evaluates the collective impact of a requirements change of the critical requirement. One of the objectives of the guideline is to limit the subjectivity and comprehensibility of the analysis of possible change impact. The guideline is based on Systems Engineering approach and its comprehensive engineering processes (Walden et al. 2015). The workflow of the method is described in Figure 4.

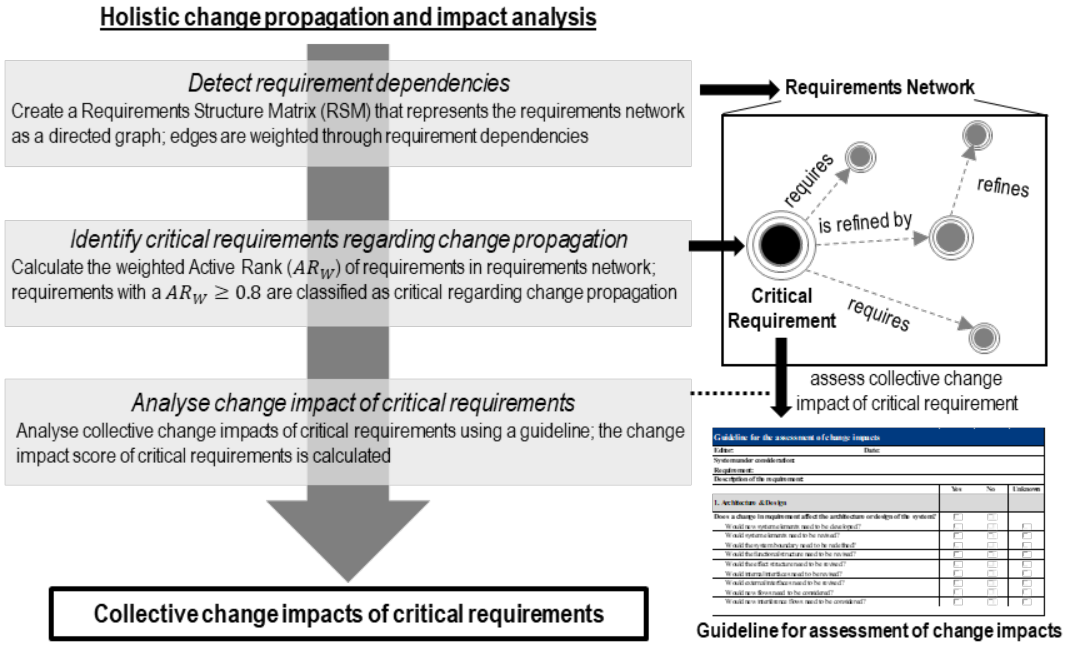

Figure 4: Workflow of holistic change propagation and impact analysis

The guide is divided into four categories: Architecture & Design, Implementation & Integration, Verification & Validation, and Other. Several questions are asked per category, e.g. “Would the functional structure need to be revised?” in Architecture & Design. Each of the questions listed in the guide can be answered either "Yes", "No" or "Unknown". The reason for the "Unknown" option is that it is not always possible to clearly determine whether a change has the effect which is discussed. In these cases, further information on the change characteristics are necessary for an unambiguous answer. The more questions answered with "Yes", the more critical the change impact. To generate a representative ratio from the given answers, "Yes" answers are assigned a value of one. They can be interpreted as the probability of occurrence of the respective effects. "No" answers are assigned the value zero. By adding up the values, the key figure for evaluating the change impact is then calculated: the change impact score. The resulting score is normalized to the value range from zero to one. Based on this score, requirements are automatically categorized, but can be adjusted by the expert manually. Categorization makes the approach compatible with overarching risk management approaches. An excerpt of the guideline is presented in Figure 5.

Guideline for the assessment of change impacts			
Editor: Date:			
System under consideration:			
Requirement:			
Description of the requirement:			
	Yes	No	Unknown
1. Architecture & Design			
Does a change in requirement affect the architecture or design of the system?	☐	☐	
Would new system elements need to be developed?	☐	☐	☐
Would system elements need to be revised?	☐	☐	☐
Would the system boundary need to be redefined?	☐	☐	☐
Would the functional structure need to be revised?	☐	☐	☐
Would the effect structure need to be revised?	☐	☐	☐
Would internal interfaces need to be revised?	☐	☐	☐
Would external interfaces need to be revised?	☐	☐	☐
Would new flows need to be considered?	☐	☐	☐
Would new interference flows need to be considered?	☐	☐	☐

Figure 5: Excerpt of guideline for the analysis of change impacts

Validation

Validation of the introduced method for holistic change propagation and impact analysis is structured according to the performance goals. For evaluation, four product examples are analysed and an industry workshop was conducted. Prior to the workshop, documentation of the holistic change propagation and impact analysis approach was provided to industry partner participants. During the workshop, an overview of the approach was given and questions were clarified. Subsequently, the fulfillment of the performance goals was discussed successively. Three participants from the first workshop attended this workshop: the head of department, a requirements manager and an engineer.

PG-1 (Only Natural language requirement specification and expert knowledge is needed.): Fulfilled

Only requirement descriptions as natural language data are required for identification of critical requirements. Expert knowledge is used to assess the collective change impact. No further input information is required.

PG-2 (Processability of complex requirement sets): Fulfilled

The processability of complex requirement sets is evaluated based on the method application with data from product example "intelligent robotic arm". The data included 50 requirements from mechanics, software and electronics/electrics. The identification of critical requirement appears to be time-efficient (processing time < two seconds) and only requires requirement priority as input, which is independent from requirement discipline. As a result, this performance goal is considered to be fulfilled.

PG-3 to 7 (Analysis of consecutive effects; Analysis of local effects; Differentiation of requirement dependency types; Consideration of higher order requirement dependencies; Analysis of collective impact): Fulfilled

The method includes all required aspects of analysis. For identification of critical requirements consecutive effects are assessed, based on the amount and type of dependencies. Higher order dependencies are included without restrictions. Additionally, the guideline enables analysis of local change effects as well as collective change impact, depending on whether the dependency network is considered (collective) or not (local). Accordingly these performances goals are fulfilled.

PG-8 (Acceptable application effort): Fulfilled
The identification of critical requirements is fully automated and does not need preprocessing of requirement data. Application effort is required for analysis of local and collective change impact. Since critical requirements are selected for expert analysis, application effort is significantly reduced in comparison to unfiltered analysis of each requirement. The guideline for local and collective impact analysis was evaluated within the industry workshop. All experts rated the application effort as reasonable and suitable for industrial application. Therefore, this performance goal is considered to be fulfilled.

PG-9 (Availability of required information and data): Fulfilled
This performance goal was evaluated as part of the industry workshop. The method and required information and data were pointed out and discussed. All experts confirmed availability of priority data and expert knowledge to assess the impact, but emphasized its spread over different experts and departments. Therefore, a group meeting for impact analysis with experts from all involved disciplines is recommended.

PG-10 (Availability of required software): Fulfilled
The availability of software is fulfilled, since no external software is required for analysis and open source code used has no regimentation regarding commercial purposes.

PG-11 (Plausibility of results): Partly fulfilled
Due to the availability of data, plausibility of results was validated separately for identification of critical requirements and change impact analysis of the critical requirements. Plausibility of results from method part one "identification of critical requirements" was validated based on requirement data from the development of an intelligent robotic arm (50 requirements) and a student racing team development project of an inverter casing (32 requirements). For both examples, requirement dependency data as well as expert knowledge on the product was available. Critical requirements resulting from application of the method were discussed with an expert involved in each development projects. Plausibility of results was confirmed by the experts, considering potential or actual change propagation effects. Plausibility is defined as reasonable from a logical perspective, since the underlying Page Rank algorithm is widely accepted and therefore mathematical correctness of results is considered as given.

Plausibility of method part two "change impact analysis of the critical requirements" was validated by an interview with two industry experts. Based on experiences from two industry examples of automotive drive train sensor development (each about 1000 requirements), plausibility of assessment was discussed and confirmed. Still, validity of results is limited. Long-term investigations based on a detailed change history including change propagation is needed in different application contexts to increase significance of results and enable analysis of accuracy. As a result, this performance goal is considered to be partly fulfilled.

PG-12 (Comprehensible results): Partly fulfilled
Comprehensibility of results was evaluated as part of the industry workshop. All experts confirmed that the results are transparent and comprehensible to a necessary extent. Limitations are indicated in the context of Page Rank calculations, which is not transparent and the Page Rank value itself cannot be interpreted, directly. Still, experts positively rated its benefit by ranking requirements and thereby reducing complexity of analysis, in comparison to unaided analysis of requirement change impact with unfiltered requirements. The systematic structure of impact

analysis guideline rated as comprehensible. As a result, comprehensibility of results is partly fulfilled with improvement potential in the context of propagation analysis.

Discussion and Outlook

This contribution introduces a method for impact analysis which addresses three major shortcomings of preceding research: First, the developed method is capable to analyse extensive interdisciplinary requirement sets of complex systems automatically and differentiate dependency types regarding propagation behavior. Second, it includes analysis of local change effects which has not been achieved systematically so far. Existing approaches only assess change propagation effects from requirement network characteristics. Enriching the analysis by local change behavior and dependency types not just increases accuracy: in combination with the high degree of automation, it opens up new research opportunities e.g. regarding change simulation, consistent change implementation and transfer to model-based impact analysis. Third, the proposed method enables preventive risk management for requirement changes in early development phases instead of reactive change treatment. Thereby, it contributes a new method to increase efficiency in development and holistic understanding of impact analysis factors and performance goals.

Key advantage of applying the method in industrial practice is providing information on change impact that enables sound decision making for development engineers at two stages:

- before a change occurs in terms of preventive measures to avoid or reduce change likelihood and impact.
- during decision process regarding change request in terms of weighting in order to accept, modify or reject change requests.

Both stages currently lack support in decision making and rely on subjective evaluations and individual skills. The introduced method provides objective and holistic information for informed decisions and requires little application effort. This helps to reduce negative change impact in development projects. Future research should aim to assess and improve accuracy of results, integrate impact analysis into development processes and enrich analysis towards further elements (e.g. functions and system elements). In addition, it is necessary to examine how the business environment must be designed to ensure organizational learning from the results of this method (Brøns Kringelum and Brix 2021).

Acknowledgments: This research was enabled by funding of the German Ministry for Education and Research (BMBF) in the context of the project ARCA ("Automated requirement change analysis for complex technical systems"), which is part of the Software Campus initiative.

References

Balzert, Helmut. 2009. Lehrbuch der Softwaretechnik: Basiskonzepte und Requirements Engineering. 3rd ed. Lehrbücher der Informatik. Heidelberg: Spektrum Akademischer Verlag.

Blessing, Lucienne T.M., and Amaresh Chakrabarti. 2009. DRM, a Design Research Methodology. London: Springer London.

Bohner, Shawn A., and Robert S. Arnold. 1996. Software Change Impact Analysis. Los Alamitos, Calif. IEEE Computer Society Press.

Brøns Kringelum, Louise, and Jacob Brix. 2021. "Critical Realism and Organizational Learning." TLO 28 (1): 32–45.

Browning, Tyson R., and Steven D. Eppinger. 2002. “Modeling Impacts of Process Architecture on Cost and Schedule Risk in Product Development.” IEEE Transactions on Engineering Management. 49 (4): 428–42.

Clarkson, P. J., Caroline Simons, and Claudia Eckert. 2004. “Predicting Change Propagation in Complex Design.” Journal of Mechanical Design 126 (5): 788–97.

Eben, Katharina G. M., and Udo Lindemann. 2010. “Structural Analysis of Requirements-Interpretation of Structural Criteria.” In DSM 2010: Proceedings of the 12th International DSM Conference, Cambridge, UK.

Gärtner, Thomas, Norbert Rohleder, Christopher M. Schlick, and others. 2008. “Simulation of Product Change Effects on the Duration of Development Processes Based on the DSM.” In DSM 2008: Proceedings of the 10th International DSM Conference, 199–208, Stockholm, Sweden.

Goknil, Arda, Ivan Kurtev, Klaas van den Berg, and Wietze Spijkerman. 2014. “Change Impact Analysis for Requirements: A Metamodeling Approach.” Information and Software Technology 56 (8): 950–72.

Graessler, Iris, Christian Oleff, and Philipp Scholle. 2020. “Method for Systematic Assessment of Requirement Change Risk in Industrial Practice.” Applied Sciences 10 (23): nr. 8697.

Gräßler, Iris, Daniel Preuß, and Christian Oleff. 2020. “Automatisierte Identifikation Und Charakterisierung Von Anforderungsabhängigkeiten – Literaturstudie Zum Vergleich Von Lösungsansätzen.” In Proceedings of the 31st Symposium Design for X (DFX2020), edited by Dieter Krause, Kristin Paetzold, and Sandro Wartzack, 199–208, Hamburg: TuTech-Verlag.

Gräßler, Iris, Henrik Thiele, Christian Oleff, Philipp Scholle, and Veronika Schulze. 2019. “Method for Analysing Requirement Change Propagation Based on a Modified Pagerank Algorithm.” Proceedings of the Design Society International Conference on Engineering Design 1 (1): 3681–90.

Hamraz, Bahram, Nicholas H. M. Caldwell, and P. J. Clarkson. 2013. “A Holistic Categorization Framework for Literature on Engineering Change Management.” Systems Engineering 16 (4): 473–505.

Hamraz, Bahram, Nicholas H.M. Caldwell, David C. Wynn, and P. J. Clarkson. 2013. “Requirements-Based Development of an Improved Engineering Change Management Method.” Journal of Engineering Design 24 (11): 765–93.

Hein, Phyo H., Nathaniel Voris, and Beshoy Morkos. 2018. “Predicting Requirement Change Propagation Through Investigation of Physical and Functional Domains.” Research in Engineering Design 29 (2): 309–28.

Hellenbrand, David, Katharina Helten, and Udo Lindemann. 2010. “Approach for Development Cost Estimation in Early Design Phases.” In DS 60: Proceedings of DESIGN 2010, the 11th International Design Conference, 779–88, Dubrovnik, Croatia.

Jayatilleke, Shalinka, and Richard Lai. 2018. “A Systematic Review of Requirements Change Management.” Information and Software Technology 93:163–85.

Koh, Y., M. Caldwell, and John Clarkson. 2012. “A Method to Assess the Effects of Engineering Change Propagation.” Research in Engineering Design 23 (4): 329–51.

Lemmens, Yves, Marin Guenov, André Rutka, Peter Coleman, and Tobias Schmidt-Schäffer. 2007. "Methods to Analyse the Impact of Changes in Complex Engineering Systems." In 7th AIAA ATIO Conf, 2nd CEIAT Int'l Conf on Innov and Integr in Aero Sciences, 17th LTA Systems Tech Conf, Followed by 2nd TEOS Forum. Reston, VA: American Institute of Aeronautics and Astronautics.

Morkos, Beshoy. 2012. "Computational Representation and Reasoning Support for Requirements Change Management in Complex System Design." Dissertation, Clemson University.

Morkos, Beshoy, James Mathieson, and Joshua D. Summers. 2014. "Comparative Analysis of Requirements Change Prediction Models: Manual, Linguistic, and Neural Network." Research in Engineering Design 25 (2): 139–56.

Neumann, Marc. 2017. "Ein modellbasierter Ansatz zur risikoorientierten Entwicklung innovativer Produkte." Dissertation, Shaker Verlag GmbH.

Oehmen, Josef, and U. Lindemann. 2016. "Risiko-Und Chancenmanagement in Der Produktentwicklung." In Handbuch Produktentwicklung, 59–97: Hanser.

The Standish Group. 2017. "Chaos Manifesto 2018.". Boston, MA, USA.

Walden, David D., Garry J. Roedler, Kevin Forsberg, R. D. Hamelin, and Thomas M. Shortell, eds. 2015. Systems Engineering Handbook: A Guide for System Life Cycle Processes and Activities; INCOSE-TP-2003-002-04, 2015. 4. edition. Hoboken, NJ: Wiley.

Wickel, Martina C. 2017. Änderungen Besser Managen. Dissertation, München: Technische Universität München.

Xing, Wenpu, and Ali A. Ghorbani. 2004. "Weighted PageRank Algorithm." Proceedings of Communication Networks and Services Research, 305–14.

Zhang, He, Juan Li, Liming Zhu, Ross Jeffery, Yan Liu, Qing Wang, and Mingshu Li. 2014. "Investigating Dependencies in Software Requirements for Change Propagation Analysis." Information and Software Technology 56 (1): 40–53.