21ST INTERNATIONAL CONFERENCE ON ENGINEERING DESIGN, ICED17
21-25 AUGUST 2017, THE UNIVERSITY OF BRITISH COLUMBIA, VANCOUVER, CANADA

A NETWORK-BASED APPROACH TO IDENTIFY LACKING COORDINATION USING HIGHER ORDER LINKS

Weidmann, Dominik; Becerril, Lucia; Hollauer, Christoph; Kattner, Niklas; Lindemann, Udo
Technical University of Munich, Germany

Abstract

Mechatronic products are getting more complex whereas disciplines becoming more interdependent in future. The coordination of discipline-spanning interfaces is going to play a more important role. Based on literature research and industry interviews this contribution identified the need for a better coordination in interdisciplinary development projects. The presented approach addresses the identification of lacking coordination using higher order links with focus on requirements and their affected stakeholders. A network-based approach takes components and functions into account to meet the characteristics of mechatronic development. Workflows can identify missing links between organizational units via the links between the artefacts. Important steps of the approach were successfully realized in an academic context, but the resulting networked based model bears potentials for further analysis and optimization. Especially an increasingly digitalized development environment requests for a software tool to use the great potentials of automatized support to analyze the system.

Keywords: Design management, Mechatronics, Requirements, Product modelling / models, Organisation of product development

Contact:
Dominik Weidmann
Technical University of Munich
Institute of Product Development
Germany
weidmann@pe.mw.tum.de

Please cite this paper as:
Surnames, Initials: *Title of paper*. In: Proceedings of the 21st International Conference on Engineering Design (ICED17), Vol. 4: Design Methods and Tools, Vancouver, Canada, 21.-25.08.2017.

1 INTRODUCTION

Over the past years product complexity increased while available time in development simultaneously decreased (Pohl and Rupp, 2015). Mechatronic design shifts towards increasing integration of information technology, which results in more complex development processes (Tomiyama et al., 2007; Komoto and Tomiyama, 2011). This increasing interdisciplinary design causes new cross-domain links and leads to increasing coordination effort. Therefore, cross-domain interfaces and requirements management are currently key aspects in mechatronic research (Ćatić and Malmqvist, 2010). Schedl (2008) identifies a high failure potential in mechatronic requirements engineering, because of partly conflicting interdependencies between the sub-disciplines mechanics, electronics and computer science. Mainly different terminologies and working methods lead to challenges concerning discipline interfaces. Hellenbrand (2013) points out the function-based mind-set in electronics and software engineering in contrast to the component-oriented mind-set in primarily mechanic disciplines. This mind-set is reflected in discipline-specific modelling as well as in either continuous (mechanics) or step wise (software/electronics) realization of functions. Consequently, requirements engineering needs to consider multi-disciplinary perspectives (Wiesner et al., 2015). Hackenberg et al. (2014) underline the suitable handling of requirements as a key factor for successful engineering processes and Moehringer (2009) again points out that interdisciplinary collaboration and communication is a key challenge in product development. Existing approaches in model-based engineering, mainly used in Aerospace applications, already address this topic. They are highly complex and mainly based on special modelling languages, which can only be used by trained professionals (Liebel et al., 2016).
To verify these challenges identified by literature research, industry interviews were conducted. The interviews confirmed the literature perspective. From an industry perspective, particularly, the connection of discipline specific types of specifications, the connection of different product levels from system to subsystem and the creation of a common understanding are currently core issues.
This contribution focus on requirements and their affected stakeholders and therefore extends and combines existing approaches by using a networked based approach. Within a Systems Engineering background and the idea of a system decomposition from requirements via functions to components, an organizational perspective is integrated. Components and functions are used to detect higher order links between requirements and particularly organizational units.
This paper is structured as follows: Section 2 briefly presents the research methodology, Section 3 gives necessary basic background knowledge about Systems Engineering, interdisciplinary requirements engineering and dependencies between development artefacts and the organizational structure. Section 4 describes the approach of this contribution and Section 5 exemplifies it within a short case. Section 6 sums up the results and gives an outlook for future research issues.

2 RESEARCH METHODOLOGY

Starting point of this research is a literature analysis in the field of mechatronic requirements engineering. This leads to further open challenges in mechatronic design, whereas a more detailed literature based clarification gives more specific insights. Existing approaches, which address the current challenges, are identified and briefly analyzed. Additionally, industry interviews are conducted in this early state of the research to specify the research topic and to ensure the practical relevance. Mainly project managers of small and medium sized companies with background in the mechatronics industry are involved in this study.
These industry insights confirmed the identified issues from literature. Existing approaches, which address this interdisciplinary coordination, are discussed. Using them as a guideline, a new approach is developed that adds further features.
The presented approach is developed and partly applied in term of an academic student project. A group of students developed an electrically driven longboard based on a previously designed and physically realized prototype. All information are documented during the project, whereas the visualization and prototypical analyses are conducted afterwards. This close to reality case gives valuable insights and is used to proof the general feasibility of the concept. A more detailed description of the case is given in Section 5.

3 BACKGROUND

3.1 Basics of Systems Engineering Methodologies

Systems Engineering already proposes a systems thinking perspective (Haskins, 2011) and describes the analogy of system and organizational hierarchies. The decomposition of systems into subsystems typically includes interdisciplinary issues involving multiple heterogeneous and distributed systems. Therefore Systems Engineering (Haskins, 2011) defines technical processes that address the product architecture and interdisciplinary requirements engineering: stakeholder requirements definition, requirements analysis, and architectural design. These processes describe the procedure from a common set of stakeholder requirements, their transformation into technical product requirements till the synthesis of a solution that satisfies the system requirements. The interdisciplinary coordination is described by various Systems Engineering methodologies. These methodologies typically cover topics like traceability between stakeholder requirements, the system's context and states and decomposition of the system. In the following, basic Systems Engineering methodologies are briefly introduced.

The Systems Modeling Toolbox (SYSMOD) provides a set of tasks with input and output, work products, guidelines and best practices. It is based on OMG Systems Modeling Language (OMG SysML) and especially addresses the modeling of requirements and the system architecture (Weilkiens, 2016). SYSMOD describes how to derive requirements from stakeholders and from the project context, as well as the requirements decomposition. During this Systems Engineering process requirements are connected to refining requirements, test cases and components of the system. An earlier described method is the FAS (Functional Architectures for Systems) that focus on the creation of functional architectures in SysML (Lamm and Weilkiens, 2010). It is a pragmatic way to derive a functional architecture from functional requirements and use cases.

Another approach, the RFLP (Requirements – Functional – Logical – Physical) approach describes a methodology to decompose a system from its requirements to functions to a logical structure to a physical structure (Kleiner and Krame, 2012). The systematic product development starts from a system analysis and ends with the physical development. It is based on the left decomposition leg of the V-model. The partial models are interconnected to enable traceability through all levels; for instance, from a function to the requirement or to the logical design. The linkages of the individual RFLP representations thus ensure constant validation and verification.

The Harmony SE approach represents another Systems Engineering methodology that focuses on integrated systems and software development processes (Hoffmann, 2011). It is again based on the V-model and the model-based systems engineering language SysML. To support the requirements analysis process Harmony SE provides two models, the requirements model and the system use case model. Whereas a requirement model visualizes the taxonomy of requirements, the system use case model groups requirements into use cases. Essential elements of this model driven development in Harmony SE are requirements documentation and requirements traceability. Consequently requirements are linked with the system's functions and the architecture.

An explicitly discipline spanning approach for Model Based Systems Engineering is CONSENS (Conceptual Design Specification Technique for the Engineering of Complex Systems)(Gausemeier et al., 2013). It consists of a modeling technique and its utilization for project planning as well as assessment and control. All partial models are in relation with each other and consequently, the principle solution consists of a coherent system of partial models that describe the concept of the product. The six partial models describe the environment, the application scenarios, the requirements, the functions, the active structure, the shape and the behavior of the system.

The presented approaches differ in the addressed systems and the specific provided support, but all these approaches have in common that they deal with the same elements like, requirements, functions and components.

3.2 Interdisciplinary Requirements Engineering

Due to this study was conducted in the background of mechatronics, the term mechatronic is briefly introduced and some focused literature is cited. The term mechatronic was introduced to describe the integration of electronics into traditional engineering (Habib, 2007). In the past, computer engineering was seen as a part of electrical engineering, but with its increasing importance a shift towards three at least equal disciplines can be observed. Silva (2005) describes mechatronics even more as the synergistic application of mechanics, electronics, controls, and computer engineering.

Requirements engineering is often partly integrated in generic development process models like the V-model. The V-Model represents a cross domain development model (Gausemeier and Moehringer, 2003), however, each discipline has its own models. Here, literature provides a great number of proposed procedures for requirements engineering, which, concerning the main tasks, mostly only differ in a few aspects (Weidmann et al., 2016). Three exemplary procedures in requirements engineering are Pohl and Rupp (2015) with its origin in software development, Haskins (2011) with an interdisciplinary point of view in the context of systems engineering, and Ponn (2011) within the domain of technical products with background in mechanical engineering. In a very simplified manner, the requirements engineering process can be broken down into four major activities: identification and elicitation, documentation, structuring and consolidation, and management.

In the field of interdisciplinary development of production systems, Schedl (2008) developed a feature based approach. This approach focuses on the improvement of mechatronic design through an integrated procedure to identify and structure requirements in the development process. It is based on existing methodological approaches for systems development and integrates approaches from the software domain to obtain a continuous use of customer requirements. Study results tend to suggest that a higher number of defined requirements predict higher project success, early definition of functional requirement relates to project success and it is important to continually evolve the requirements throughout the project (Summers et al.). Szejka et al. (2014) examine the compatibility of requirements of three different dimensions in complex systems. Therefore, they discuss requirements management based on the cross-domain dimension, the lifecycle phase, and the requirements interoperation dimension. They underline the risk of inconsistency when all three phases are involved and present an overview of existing gaps, which raises a discussion about inconsistency in requirement across the dimensions. To face the challenge of inconsistencies caused by the interdisciplinary nature of mechatronic design, Politze and Bathelt (2009) developed a method to exploit functional product requirements. They start from the point that the quality of a product is often judged by the quality of its functions. Thus, they extended the traditional requirements list in the early design stages with functional requirements. Typically, requirements documentation in small and medium sized companies is based on tables, whereas common types provide either an object-oriented structure (Mayer-Bachmann, 2007) or a feature-oriented structure (Politze and Bathelt, 2009). However, such a list does not support the identification and management of interfaces. For these purposes additional model-based techniques like Impact Network Model or Consistency Matrix can be used. Politze and Bathelt (2009) present an approach to directly benefit from a function oriented product description by deriving extended function structures. This new structure explicitly includes sensors, actors and the control logic.

Assar (2014) gives a brief overview over the state of the art in Model Driven Requirements Engineering. He analyzed 29 approaches according to three main criteria: research issue, research contribution and evaluation method. This analysis shows that most approaches deal with new languages for requirements representation and derivation of system specifications, whereas aspects like requirements elicitation and requirements validation methods are much less considered. Traceability issues are only rarely discussed.

3.3 Dependencies of Requirements, Product Architecture and Organizational Units

Mechatronic development is an interdisciplinary approach that has to deal with the inherent complexity of its products. As mentioned in the both previous sections, many existing approaches address the interdisciplinary development. However, many small- and medium sized companies still struggle with a lack of transparency regarding interfaces on product, organizational or process level (Chucholowski et al., 2016). The following approaches mostly exclude the process level, however, they often deepen the product level. The basic elements on product level are similar to the ones addressed in Systems Engineering: requirements, functions and the product structure.

In this context Zheng et al. (2016) describe a multidisciplinary design methodology for mechatronic systems based on an interface model. Their approach focus on the support of specific design phases in which designers structure design sub-tasks and proceed and react in unforeseen situations. Therefore, they use a two level approach with an extended V-Model at macro level and requirements, functions, and architecture on a micro level. To ensure the consistency and traceability between the two levels, the UML class diagram of multidisciplinary interface model is proposed. In order to obtain an as complete as possible overview over the influence of requirements dependencies on the development process

Zhang et al. (2005) describe a feature-oriented approach to model dependencies between requirements in context of software projects. This approach uses product hierarchies and expand them with graphical elements to visualize additional information about the dependencies of features, whereas features represent a row of closely linked customer requirements. On a concrete level, requirements dependencies are described as decomposition, characterization or specialization. Morkos et al. (2012) link requirements and use these dependencies to predict requirement change propagation. Therefore they use higher order Design Structure Matrices, because a case study revealed second order relationships, which hardly could be predicted by the engineers. Unforeseen propagation rarely occurred in first order form, rather in second order form.
On the product level, the product architecture combines two essential perspectives, the function- and object-oriented perspective. Product functions are linked with physical components and modules of the product, thus two essential parts of the system are combined in one model (Pahl and Beitz, 2013). Besides these perspectives, the product architecture can integrate an organizational perspective by showing areas of responsibility, no matter if they refer to functions or components. This is a first step to integrate the organizational structure into the technically-driven architecture model.
The structure of information flow networks is the heart of large-scale product development efforts (Braha and Bar-Yam, 2007). In context of information flows in product development an in-depth study could be found in Braha and Bar-Yam (2007). They use statistical properties of strategically important organizational networks of people, engaged in distributed product development, to provide insight into ways of improving the strategic and operational decision making of the organization. An approach to foster communication of different disciplines in mechatronic design is the functional modeling compiler. Canedo and Richter (2014) focus on the architectural design space exploration of mechatronic systems. It enhances the communication for requirements negotiation among engineers and organizations to enable multi-disciplinary simulations. The functional modeling compiler finally provides a way to evaluate the impact of domain-specific design decisions on system-level. On the one hand the method creates a technology-independent description of the system functions, on the other hand it synthesizes technology-dependent solutions to directly implement the architectural design space exploration. The dependency between the product and the organizational architecture can be described by the mirroring hypothesis (Colfer and Baldwin, 2010). It pronounces a correspondence between the organizational patterns of a development project, such as communication links, geographic collocation, and team and the technical patterns in the system. In consequence this means that poorly linked developers will design independent system components, while highly crosslinked developers will design highly interdependent system components. In further research the mirroring hypothesis could be strengthened with additional cases (MacCormack et al., 2012).
Sosa *et al.* (2004) address also the alignment of product and organizational structure. They start from the point that product development picks up existing architectures while this hinders organizations to implement novel architectures. Therefore, they investigated the impact of organizational and system boundaries, design interface strength, indirect interactions, and system modularity on the alignment of design interfaces and team interactions. The results show that the chance of misalignment is greater across organizational and system boundaries, but boundary effects may also affect weak and strong interfaces. In further research they came out with a project management tool based on the design structure matrix that should support companies to identify potential failures in planned communication (Sosa et al., 2007). Therefore, a design interface matrix puts components in relation and a team interaction matrix puts teams in interaction. Both matrices are combined to an alignment matrix that reveals matched interfaces, unattended interfaces, and unidentified interfaces. Hence, critical communication issues can be identified and addressed by tasking teams to talk to each other.
From a more technical perspective Beier (2014) underlines the difficulty to fully understand interdependencies between elements on product level and the potential to support development by making these interdependencies explicit. The focus is on traceability and two major obstacles: high effort and unclear benefits. To address these challenges he proposes innovative concepts for the usage of traceability, i.e. by the integration into established methods. Furthermore, he examined the visualization of this information to support developers in handling complex systems and developed a concept for visualizing artefact-spanning traceability information. This concept links requirements, functions and the product structure. An approach that links design artefacts on product, organizational and process level is presented by Chucholowski et al. (2016). In context of systematic partitioning of mechatronic products they extend and combine existing approaches and integrate domain and discipline

allocations based on their structural dependencies. On organizational level departments or individuals are included and on product level functions and components are considered, like in the product architecture. Additionally, working principles are included and dependencies and hierarchies in each system and among the elements are considered. Here, disciplines can be represented by the organizational system.

4 A NETWORK-BASED APPROACH TO IDENTIFY LACKING COORDINATION USING HIGHER ORDER LINKS

The previous sections present approaches that deal with processes and procedures to systematically develop interdisciplinary products from requirements via functions into physical elements. Moreover, they point out the importance of interdisciplinary communication and cooperation as well as the barriers of system or organizational boundaries. Existing approaches connect elements of different development phases (i.e. requirements, functions, and components) and focus a consistent system with continuous traceability. Other approaches, mostly with background in business research, integrate an organizational perspective, either to identify communication gaps or to align organizational and product structure.

This contribution clearly focus on requirements and their affected stakeholders. The approach extends and combines existing approaches by using a networked based approach and taking domain and discipline allocations (organizational and technical boundaries) into account based on structural dependencies. The approach improves interdisciplinary coordination on requirements level during the product development and thus supports a continuous specification of requirements. Further issues are the identification and management of domain-spanning interfaces and relations. Within a Systems Engineering background and the idea of a systems decomposition from requirements via functions to components an organizational perspective is integrated. The approach is particularly based on the ideas of Beier (2014), Chucholowski et al. (2016), and Sosa et al. (2007) and the interconnection of product artefacts.

Figure 1 shows the underlying connection logic. Requirements can be directly linked to organizational units (as source or responsibility), functions (functional requirements), and components (non-functional requirements). Components and functions are connected by the product architecture and the organizational units can be assigned to either functions (i.e. software development) or components (i.e. mechanical engineering).

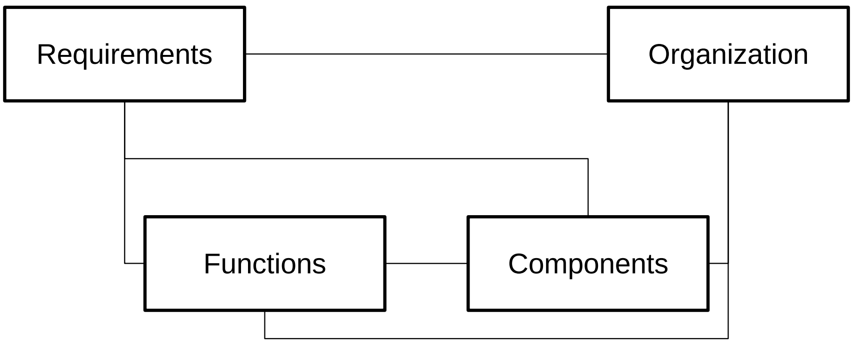

Figure 1. Dependencies of elements

As shown in Sosa et al. (2004) first order links are relatively easy to handle, however, higher order links are challenging. Sosa et al. (2007) put the organizational structure and components in context using design structure matrices. This contributions uses components and function to detect higher order links between requirements and particularly organizational units. A schematic representation of the approach, its goals and the effects is shown in Figure 2. The system with the above defined elements and its links is analyzed. First all relevant stakeholders connected with a requirement can be identified: R2 is indirectly linked to S1 and S2. Secondly, new cross discipline links could be identified: R2 and R3 are indirectly linked to M1, S1 and S2. At this point it becomes necessary to differ between forward and backward traceability. It is useful to integrate a backward traceability to trace a requirement to its source; in this approach, however, we focus on the forward traceability, meaning which responsibilities (e.g. persons, departments, and domain) are linked with which elements. Responsibilities can be assigned to functions, product objects or requirements itself, depending on a company's specific distribution of

responsibilities. In contrast to previous approaches, this approach connects requirements with stakeholders via indirect links in a network-based structure. Hence, discipline and domain spanning dependencies can be identified for example to specify or change a requirement.
This network-based approach can be supported by a set of rules that automate the completion of the model and thus reveal the missing links. A prototype of a tool that implements this approach is presented in Weidmann and Lindemann (in press).

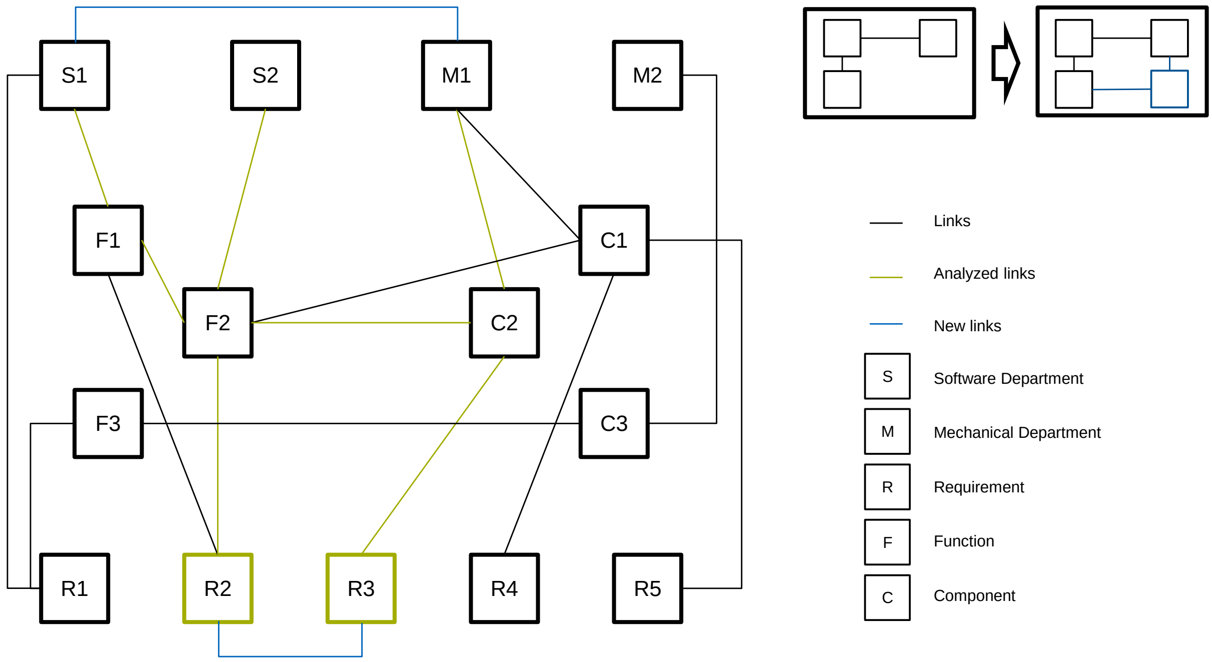

Figure 2. Identification of lacking connections using higher order links

5 CASE

5.1 Description

A first case study to verify the feasibility of the concept is conducted within a student project. Based on previously designed and physically realized prototypes, a group of students of the Technical University of Munich want to professionalize their idea and develop an electrically driven longboard. The development of the product itself didn't start from scratch, because it is based on previously existing prototypes and related findings. Hence, basic components were already defined, but requirements and functions were added afterwards. The system "board" consists of six main structural components and four core functions. Both, components and functions are further divided into subcomponents and sub features. The links between functions and components enable a systematic assignment of requirements to both, functions and components, independently how they were initially identified. Requirements which concern a module or assembly were assigned to each of the subcomponents. The visualization is implemented in Soley Studio (Soley GmbH) (Figure 3), which is basically a tool to combine data of various sources and formats and implements it into a graph model. The visualization support could not be applied during the project itself and was implemented retrospectively.

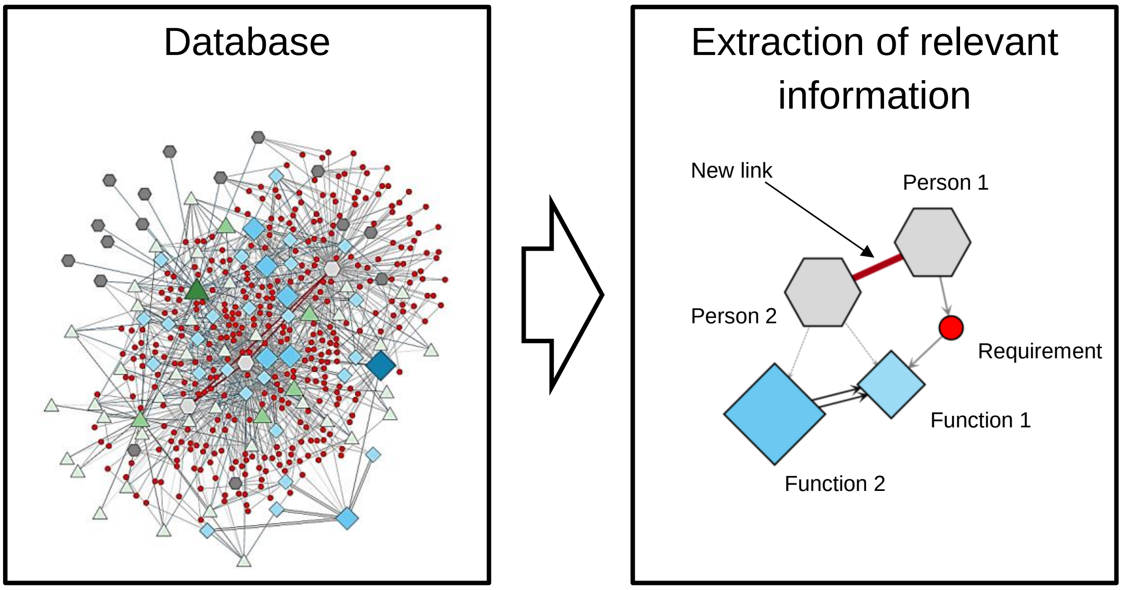

Figure 3. Left: The model consists of 445 nodes and 1176 edges created with Soley Studio (Soley GmbH); Right: Analysis of the system that extract relevant information to identify a missing link

5.2 Discussion

Due to the fact that the visualization support was generated after the project, no project accompanying analysis was conducted. The current state of the visualization only provides a representation of the actual system state. The connection of the elements were implemented, hence, the traceability in the system could be graphically highlighted. Consequently, a clear picture of affected elements could be derived, but no rules or workflows to identify the lacking links are yet implemented. Hence, the new link in the Figure 3 is only implemented manually to visualize the future idea of an automatized tool. But already this schematic representation immediately raises questions like up to which order a consideration of links is reasonable or how could the essential missing links be prioritized.

All elements, links, analyses and changes were implemented manually, which in consequence led to a high effort to create and maintain the system. Furthermore, this manual maintenance is a very error-prone process. Nevertheless, a future automatization in a tool has great potential, especially in an increasingly digitalized development environment. The networked based model offers great chances to automatize workflows and rules to derive necessary information. Moreover, a suitable visualization in combination with automated workflows bears great potential to support non experts in using the approach. Consequently, once defined and implemented, it could be applied without any specific expertise.

The approach considers the inherent properties of a mechatronic development, thus an interdisciplinary and interconnected system development. It addresses the discipline's different mind-set (Hellenbrand, 2013) by linking the organizational structure to both, functions and components.

6 CONCLUSION AND OUTLOOK

Products are getting more complex and disciplines in mechatronic development becoming more interdependent in future. Interdisciplinary coordination of discipline-spanning interfaces is going to play a more important role. Based on literature research and industry interviews this contribution identified the need for a better coordination in interdisciplinary development projects. This contribution addresses the identification of lacking connections using higher order links, whereas the focus is clearly on requirements and their affected stakeholders. Functions and components are used as higher order links to meet the characteristics of a mechatronic development. This approach extends and combines existing approaches by using a networked based approach that takes discipline characteristics (organizational and technical boundaries) into account based on structural dependencies.

Important steps of the approach were successfully realized in an academic context. The resulting networked based model bears potentials for further analysis and optimization, especially in an increasingly digitalized development environment. Therefore, the implementation into a software tool would offer great potentials in automatized support to analyze the system. The implementation of rules or workflows could standardize the analysis and handle the complexity. This is urgently necessary in context of more complex systems. Based on these automated processing and a suitable visualization the tool could support non experts in managing the interdisciplinary interfaces. A first prototype of the tool is presented in Weidmann and Lindemann (in press). Further potentials of this approach and a resulting tool would be the integration of additional elements, like test cases or various analyses, like completeness checks. Finally, a full application in an industrial context would be useful to gain deeper insights about the method and their applicability.

REFERENCES

Assar, S. (2014), “Model Driven Requirements Engineering: Mapping the Field and Beyond”, in Moreira, A. (Ed.), *2014 IEEE 4th International Model-Driven Requirements Engineering Workshop (MoDRE): 25 Aug. 2014, Karlskrona, Sweden ; co-located with the 22nd IEEE International Requirements Engineering Conference (RE'14),* IEEE, Piscataway, NJ.

Beier, G. (2014), *Verwendung von Traceability-Modellen zur Unterstützung der Entwicklung technischer Systeme, Berichte aus dem Produktionstechnischen Zentrum Berlin*, Fraunhofer-Verl., Stuttgart.

Braha, D. and Bar-Yam, Y. (2007), “The Statistical Mechanics of Complex Product Development. Empirical and Analytical Results”, *Management Science*, Vol. 53 No. 7, pp. 1127–1145.

Canedo, A. and Richter, J.H. (2014), “Architectural Design Space Exploration of Cyber-physical Systems Using the Functional Modeling Compiler”, *Procedia CIRP*, Vol. 21.

Ćatić, A. and Malmqvist, J. (2010), “Requirements Management when Introducing new Mechatronic Sub-Systems - Managing the Knowledge Gap”, *International Design Conference - DESIGN 2010*.

Chucholowski, N., Kriegler, J., Hollauer, C., Kattner, N., Becerril, L., Weidmann, D. and Lindemann, U. (2016), “Systematic Partitioning in Mechatronic Product Development by Modeling Structural Dependencies”, *18th International Dependency and Structure Modeling Conference, DSM 2016*.

Colfer, L. and Baldwin, C. (2010), “The Mirroring Hypothesis: Theory, Evidence and Exceptions”, *Harvard Business School Working Paper*, Vol. 2010.

Gausemeier, J., Gaukstern, T. and Tschirner, C. (2013), “Systems Engineering Management Based on a Discipline-Spanning System Model”, *Procedia Computer Science*, Vol. 16, pp. 303–312.

Gausemeier, J. and Moehringer, S. (2003), “New Guideline VDI 2206 – A Flexible Procedure Model for the Design of Mechatronic Systems”, *International Conference on Engineering Design ICED 03*.

Habib, M. (2007), “Mechatronics - A unifying interdisciplinary and intelligent engineering science paradigm”, *IEEE Industrial Electronics Magazine*, Vol. 1 No. 2.

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.

Haskins, C. (2011), *Systems engineering handbook: A guide for system life cycle processes and activities,* Version 3.2.2, International Council of Systems Engineering, San Diego, Calif.

Hellenbrand, D. (2013), *Transdisziplinäre Planung und Synchronisation mechatronischer Produktentwicklungsprozesse*, Dr. Hut, Munich.

Hoffmann, H.-P. (2011), *Deskbook: Model-Based Systems Engineering with Rational Rhapsody and Rational Harmony for Systems Engineering*, Vol. 2011.

Kleiner, S. and Krame, C. (2012), “Entwerfen und Entwickeln mit Systems Engineering auf Basis des RFLP-Ansatzes in V6”, *:em engineering methods AG*.

Komoto, H. and Tomiyama, T. (2011), “Multi-disciplinary system decomposition of complex mechatronics systems”, *CIRP Annals - Manufacturing Technology*, Vol. 60 No. 1.

Lamm, J.G. and Weilkiens, T. (2010), “Functional Architectures in SysML”, *Tag des Systems Engineering*, pp. 109–118.

Liebel, G., Marko, N., Tichy, M., Leitner, A. and Hansson, J. (2016), “Model-based engineering in the embedded systems domain. An industrial survey on the state-of-practice”, *Software & Systems Modeling*.

MacCormack, A., Baldwin, C. and Rusnak, J. (2012), “Exploring the duality between product and organizational architectures. A test of the “mirroring” hypothesis”, *Research Policy*, Vol. 41 No. 8, pp. 1309–1324.

Mayer-Bachmann, R. (2007), *Integratives Anforderungsmanagement: Konzept und Anforderungsmodell am Beispiel der Fahrzeugentwicklung*, Univ.-Verlag Karlsruhe.

Moehringer, S. (2009), “Mechatronic Design - Historical Background and Challenges for the Future”, *International Conference on Engineering Design, ICED'09*.

Morkos, B., Shankar, P. and Summers, J.D. (2012), “Predicting Requirement Change Propagation Using Higher Order Design Structure Matrices: An Industry Case Study”, *Clemson Engineering Design Applications and Research (CEDAR)*, Vol. 2012.

Pahl, G. and Beitz, W. (2013), *Engineering design: a systematic approach*, Springer Science & Business Media.

Pohl, K. and Rupp, C. (2015), *Requirements engineering fundamentals: A study guide for the certified professional for requirements engineering exam,* 2nd ed., Santa Barbara, CA.
Politze, D.P. and Bathelt, J. (2009), “Exploitation Method for Functional Product Requirements - An Integrated Function Oriented Approach”, *Proceedings of the 2009 IEEE International Conference on Mechatronics*.
Ponn, J. (2011), *Konzeptentwicklung und Gestaltung technischer Produkte: Systematisch von Anforderungen zu Konzepten und Gestaltlösungen,* 2nd, Springer, Heidelberg.
Schedl, S. (2008), “Integration von Anforderungsmanagement in den mechatronischen Entwicklungsprozess”, Technical University of Munich, Munich, 2008.
Silva, C.W. de (2005), *Mechatronics: An integrated approach*, CRC press, Boca Raton.
Soley GmbH, “Soley Empowering Experts”, available at: https://www.soley.io/en/ (accessed 6 April 2017).
Sosa, M.E., Eppinger, S.D. and Rowles, C.M. (2004), “The Misalignment of Product Architecture and Organizational Structure in Complex Product Development”, *Management Science*, Vol. 50 No. 12, pp. 1674–1689.
Sosa, M.E., Eppinger, S.D. and Rowles, C.M. (2007), “Are Your Engineers Talking to One Another When They Should?”, *Harvard Business Review*, pp. 133–142.
Summers, J.D., Joshi, S. and Morkos, B., “Requirements Evolution: Relating Functional and Non-Functional Requirement Change on Student Project Success”, pp. V003T04A002
Szejka, A.L., Aubry, A., Panetto, H., Júnior, O.C. and Loures, E.R. (2014), “Towards a Conceptual Framework for Requirements Interoperability in Complex Systems Engineering”, in Meersman, R., Panetto, H., Mishra, A., Valencia-García, R., Soares, A.L., Ciuciu, I., Ferri, F., Weichhart, G., Moser, T., Bezzi, M. and Chan, H. (Eds.), *On the Move to Meaningful Internet Systems: OTM 2014 Workshops*, *Lecture Notes in Computer Science*, Vol. 8842, Springer Berlin Heidelberg, Berlin, Heidelberg.
Tomiyama, T., D’Amelio, V., Urbanic, J. and ElMaraghy, W. (2007), “Complexity of multi-disciplinary design”, *CIRP Annals-Manufacturing Technology*, Vol. 56 No. 1.
Weidmann, D., Kattner, N., Hollauer, C., Becerril, L., Chucholowski, N. and Lindemann, U. (2016), “Methods Collection to Support Requirements Engineering with Focus on Structuring and Consolidation of Requirements”, *IEEE International Conference on Industrial Engineering and Engineering Management (IEEM)*.
Weidmann, D. and Lindemann, U. (in press), “Manage Interdisciplinarity Based on Reequirements Traceabilty”, *PICMET '17 Conference*.
Weilkiens, T. (2016), *SYSMOD - The Systems Modeling Toolbox, 2nd edition*, MBSE4U.
Wiesner, S., Hauge, J.B. and Thoben, K.-D. (2015), “Challenges for Requirements Engineering of Cyber-Physical Systems in Distributed Environments”, in Umeda, S., Nakano, M., Mizuyama, H., Hibino, H., Kiritsis, D. and Cieminski, G. von (Eds.), *Advances in Production Management Systems: Innovative Production Management Towards Sustainable Growth*, *IFIP advances in information and communication technology*, Vol. 460, Springer International Publishing, Cham.
Zhang, W., Hong, M., & Zhao, H. (2005), “A Feature-Oriented Approach to Modeling Requirements Dependencies”, *IEEE Computer Science*.
Zheng, C., Hehenberger, P., Le Duigou, J., Bricogne, M. and Eynard, B. (2016), “Multidisciplinary design methodology for mechatronic systems based on interface model”, *Research in Engineering Design*.