PROCEDURE MODEL FOR STRUCTURED RELATIONAL MODELING OF REQUIREMENTS TO SUPPORT REQUIREMENTS-ORIENTED DECISION MAKING

Horber, Dennis;
Schleich, Benjamin;
Wartzack, Sandro

Friedrich-Alexander-Universität Erlangen-Nürnberg

ABSTRACT

The development of complex technical systems is characterized by a large number of system elements as well as their interactions. With regard to requirements management, many requirements have to be considered, which can have different relations to each other. If these requirements are used as basis for criteria in the decision making process, these relations must also be considered in the multi-criteria evaluation of product alternatives. Therefore, a computer-aided approach is presented in this paper, which allows the systematic modeling of requirement interactions focusing on multi-criteria decision making. For this purpose, basic relation types are identified, which are used to model submatrices in order to derive the Requirement Relation Matrix (RRM). Matrix-based as well as graph-based visualization methods are used for the RRM in order to improve the alternatives with the knowledge about the relational linkage. In addition, the effects of changes in requirements can be transferred to the decision making process. The approach is exemplarily applied to the extension of a test laboratory by a test bench.

Keywords: Decision making, Requirements, Dependency, Graph visualization, Product modelling / models

Contact:
Horber, Dennis
Friedrich-Alexander-Universität Erlangen-Nürnberg
Engineering Design
Germany
horber@mfk.fau.de

Cite this article: Horber, D., Schleich, B., Wartzack, S. (2021) 'Procedure Model for Structured Relational Modeling of Requirements to Support Requirements-Oriented Decision Making', in *Proceedings of the International Conference on Engineering Design (ICED21),* Gothenburg, Sweden, 16-20 August 2021. DOI:10.1017/pds.2021.505

1 INTRODUCTION

In the development process of technical systems, the requirements elicitation is one of the first steps that is used to prepare the further product development (Rupp *et al.*, 2009). The complexity of technical systems basically consists of the number of individual system elements and their high degree of interconnection as well as their interaction with each other. In terms of requirements management, many requirements with a high degree of networking must therefore be taken into account. An important factor is the completeness of the requirement model, which has a potential influence on the success or failure of the project (Kamata and Tamai, 2007). The research field of requirements interaction management deals with these interdependencies between requirements and provides support for the discovery and management of critical relationships (Robinson *et al.*, 2003). If requirements interact with each other, this is not particularly problematic, but according to Dahlstedt and Persson (2005), they can have significant consequences for the entire product development, for example in various decision situations. In order to consider this influence of requirements among each other, use of these relations has to be made throughout product development. However, common software solutions do not provide functionalities that allow the comprehensive consideration of requirement relations and their visualization (Heim *et al.*, 2008). Decision-making support is therefore made more difficult because there is no suitable procedure or software support. According to Karlsson *et al.* (1997), this lack of support is one of the main tasks for research topics in this area, because especially in complex system developments with a huge number of requirements, support of the developer is essential. In a requirement model, many different requirement relations can exist. Carlshamre *et al.* (2001) state, that only 20% of all requirements are responsible for 75% of the interdependencies. Hence, it is necessary to identify these highly linked requirement relations as well as less linked ones in order to enable their specific consideration throughout product development. Therefore, an approach is presented in this paper that allows the systematic identification of relations between requirements and enables developers to use these relations in decision making.

2 STATE OF THE ART

At the beginning of a development project, the various boundary conditions as well as limitations for the project are defined by specifying goals and requirements, as described in the systematic design process model by Pahl *et al.* (2021), requirements serve as a metric for determining the achievement of the defined development goals. From this goal system, criteria can be derived for decision-making, with which decisions can be made on the basis of comparable values. The requirements model is constantly expanded and detailed with new information throughout the development process and there may also be dependencies between the elements. The system design methodology axiomatic design (Suh, 1990) details such relations and introduces the mathematical relationship between functional requirements and design parameters. A necessary premise is the fulfilment of the independence axiom, which must be fulfilled in uncoupled or decoupled designs and requires an equal number of functional requirements (FRs) and design parameters (DPs) in an ideal design (Suh, 1990). In Axiomatic design, the relations between FRs and DPs are modelled in the design matrix A, which can be used to develop the system and make decisions based on the modelled relations between FRs and DPs. Another approach for considering relations in product development and decision making is the House of Quality (Sullivan, 1986). Here, customer desires are weighted according to their importance and linked with the relevant engineering characteristics. In decision making, the house of quality enables designers to compare alternatives or products from competitors.

Particularly in the development of complex technical systems, this process results in multiple requirements that have to be taken into account in order to be implemented in the final product. The consideration of all requirement elements and their implementation by system elements is the subject of research in the field of requirements traceability (Dahlstedt and Persson, 2005). So-called tracelinks are modeled between the requirement and the system elements that implement the corresponding requirements. This allows developers to analyse how these requirements were implemented and with which system element it is achieved. This link therefore enables an automatable overview about the implementation of the contained requirements in a project. However, in many cases the requirements cannot be implemented in a random order, since limited capacities restrict the simultaneous implementation of all requirements. Therefore, requirements in software development are prioritized, which is called release-planning (Karlsson *et al.*, 1997). Karlsson *et al.* (1997) present a cost-value

https://doi.org/10.1017/pds.2021.505 Published online by Cambridge University Press

approach, where they determine relations regarding implementation costs as well as the value added by pairwise comparison and model them between the requirements. This allows the implementation sequence to be determined. Another application of requirement relations is the identification of effects caused by requirement changes (Morkos *et al.*, 2012). More generally than these applications, requirements interaction management considers the potential relations between requirements on a more generic level and addresses further research and novel developments of possible applications of relational requirements models (Robinson *et al.*, 2003). It describes the steps for the collection and development of relational requirements models and provides assistance for the consideration of interactions between requirements in product development (Robinson *et al.*, 2003). For example, Robinson and Volkov (2007) use this approach to identify and resolve conflicts between requirements.
However, an important prerequisite for the consideration of relations is a suitable visualization method as well as their analysis. A common modeling and visualizing method is matrix-based requirements modeling, which is often supported by a graph-based representation due to high numbers of system elements in complex systems and therefore an improved clarity (Heim *et al.*, 2008; Morkos *et al.*, 2012). Eben and Lindemann (2010) present an approach how graph-based requirement relations can be evaluated by characteristic values. Heim *et al.* (2008) state that existing software solutions regarding requirements management are unsuitable for such use and therefore new solutions have to be developed to support product developers. Especially in the development of complex systems with many requirement elements, relational modeling approaches reach their limits because of the manual modeling effort. For this reason, different approaches deal with an automated identification of relations (an extensive comparison of approaches can be found in Graeßler *et al.* (2020)). According to Graeßler *et al.* (2020) and Dermeval *et al.* (2016), ontology-based approaches are a promising way to support the identification of interactions, as presented exemplary by Soomro *et al.* (2014).
However, dependencies and interactions do not only occur in the area of requirements management, but also in the area of decision making, although in multi-criteria evaluation a condition is that criteria are independent of each other (Breiing and Knosala, 1997). Regarding this restriction, the Analytic Hierarchy Process (AHP) was developed, which was then extended to the Analytic Network Process (ANP), which consider the interactions occurring in practice (Saaty, 2004; Sipahi and Timor, 2010). Relationships between criteria therefore exist and must be taken into account (Carlsson and Fullér, 1995). Öztürk (2006) provides a comprehensive analysis of existing decision making processes with a consideration of criteria dependencies. Although these approaches use relations between criteria, they are captured and modeled in a problem-oriented way during decision making. A procedure that combines the approaches from requirements management, requirements interdependency management and multi-criteria decision making does not exist so far.

3 RESEARCH QUESTIONS

Based on the presented state of the art it can be seen that the modeling of relations between requirements makes an important contribution to capture the complexity of the system and to represent it in the requirements model. Many approaches focus on the traceability of requirements, which enables the verification of the implementation of requirements. An important application is the impact prediction of requirement changes. These functions are already realized by existing software solutions today. In the area of decision making, relations between evaluation criteria often play a subordinate role, since in many procedures the independence of criteria is required (Saaty, 2004). Nevertheless, there are approaches that can take dependencies into account (see (Öztürk, 2006)). Relationships can, however, also exist between these two partial models, especially when criteria are based on requirements. Thus, requirement relations can be potentially transferred to criteria and taken into account within decision-making. However, there is no suitable approach or procedure for this. The advantage of such an approach is the possibility of context-specific analysis during decision making by analyzing the dependencies between the model elements. Therefore, this paper deals with the question of how requirements and criteria derived from them can be related to each other in a structured way and how these relations can be modeled (1). In addition, the question is answered, how to use these relations in the decision making process (2).

https://doi.org/10.1017/pds.2021.505 Published online by Cambridge University Press

4 STRUCTURED RELATIONAL MODELING OF EVALUATION CRITERIA

When modeling requirements, knowledge of relations between the individual elements is an essential aspect for the representation of interactions at the requirement level. If the modeled requirements serve as a basis for the criteria model of a multi-criteria decision problem, these relations can be transferred to the criteria model. Before presenting the developed approach, the present contribution is first classified into the development process as well as requirements-oriented decision making in section 4.1. Subsequently, possible relations between requirements are derived from the literature in section 4.2. Based on the resulting relation categories, section 4.3 presents the modeling of partial models as well as the procedure for relational modeling between requirements and criteria. In the following section 4.4 the presented approach is applied to an exemplary use case, for which a demonstrator regarding modeling and visualization of relations was developed. The present contribution closes with a critical discussion of the results in section 5 and a summary as well as an outlook in section 6.

4.1 Requirements-oriented decision making in product development

In the development of complex systems, the objectives as well as the requirements of the project are first defined at the beginning of development. Different stakeholders have an interest in the development project and place different demands on the system to be developed (Rupp *et al.*, 2009). During product development alternatives are developed, from which the most suitable one has to be selected and realized further. For this purpose, multi-criteria decision making is used, in which the alternatives are evaluated on the basis of various criteria, for example with AHP (Saaty, 2004). Therefore, it is necessary to transfer the requirements into a criteria model, resulting in decision criteria representing those requirements in order to make decisions according to the requirements model. However, it is common that criteria are defined only shortly before the decision situation, which can lead to criteria not corresponding to the requirements, nevertheless because of the manual definition. Horber *et al.* (2020) therefore present an approach that allows the (semi-) automated derivation of requirements into evaluation criteria already during the requirements specification (see Figure 1). This approach is therefore understood in this paper as a *requirements-oriented decision making*.

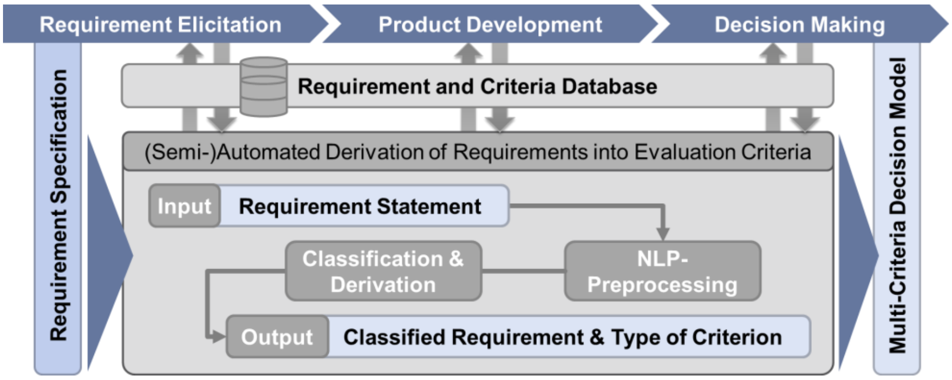

Figure 1. Proposed approach for requirements-oriented decision making based on a (semi-) automated derivation of evaluation criteria by Horber et al. (2020)

4.2 Requirement and criteria relations

Modeling requirements and the automated derivation into evaluation criteria allows decision makers to take all relevant requirements into account. Knowledge about the dependencies and interactions between requirements is an essential component for the integration of system complexity into the requirements model. The direct connection of the evaluation model with the requirement model enables the transfer of the system complexity into the criteria model as well. Due to a consistent derivation of requirements into evaluation criteria, modeled interactions between requirements apply also between evaluation criteria and can be further used for the support of decision situations. For example, during decision making process, where one concept from multiple concept alternatives has to be chosen for further development, the knowledge about the interactions can be used to improve those concepts regarding limiting characteristics in terms of evaluation results. However, it has to be examined first, which basic types of relations may occur and which relation types can be used in the developed approach. Reviewing the literature on requirement relations, it shows, that the consideration

https://doi.org/10.1017/pds.2021.505 Published online by Cambridge University Press

of interactions between requirements is primarily applied in software development. In a comprehensive study, Dahlstedt and Persson (2005) identify different types of interdependencies and present a model for fundamental interdependency types that improves requirement traceability. In order to improve the effort for development of new relational requirements models, Ramesh and Jarke (2001) present a reference model that considers basic traceability-links between requirements and therefore focuses on traceability as well. Another goal is pursued by Karlsson *et al.* (1997), who use relations in their developed cost-value approach to prioritize requirements in software planning. Carlshamre *et al.* (2001) use these relations in their industrial survey and extend them to capture the occurrence of these relations in different companies and their projects. A comprehensive work is provided by Robinson *et al.* (2003), who presents the fundamentals of requirements interaction management and consider the relations between requirements in very general manner. They describe the activities of discovery, management, and disposition of critical relationships (Robinson *et al.*, 2003).
Breiing and Knosala (1997) present a very generic model from the research field of decision making, which contains only the most important relations necessary for multi-criteria evaluation. A summary of the analysis is shown in Table 1.

Table 1: Comparison of types of relations between requirements

Author(s)	**Relation family**	**Relation type**	**Category**
Dahlstedt and Persson (2005)	Cost	Increase/Decreases_cost_of	Cost
	Value	Increase/Decreases_value_of	Value
	Constrain	Requires, Conflicts_with	Relationship
	Structural	Refined_to, Change_to, Similar_to	Structure
Karlsson *et al.* (1997)	Existence	Cannot-exist, Must-exist	Relationship
	Cost	Positive Cost, Negative Cost	Cost
	Value	Positive Value, Negative Value	Value
Ramesh and Jarke (2001)	Traceability Links	Satisfaction, Evolution, Rationale, Dependency	Dependency, Satisfaction
Breiing and Knosala (1997)	Requirement Relations	Independent, supportive, contrary, conflicting	Relationship
Robinson *et al.* (2003)	Interaction Type	Positive, Negative, Unspecified, No	Dependency, Relationship
	Basic Interactions	Structure, Resource, Task, Causality, Time	Structure
Carlshamre *et al.* (2001)	Interdependencies	And, Or, Requires, Temporal, CValue, ICost	Condition, Value, Cost

Based on the literature presented, different relation families and relation types were identified. Within the relation types seven different categories could be identified (see Table 1), which are derived as relevant relations between requirements or evaluation criteria in the presented approach. *Dependency* describes whether there is a basic interaction between two requirements (*dependent*) or whether no interaction can be perceived (*independent*). Therefore, it is not of interest, what value this type of relation has. Based on this fundamental relation, *Condition* describes those relations, which signalize requirements that belong together (*and*), exclude each other (*xor*) or are alternative (*or*). The relations of type *Structure* are used to structure two similar requirements with different information content. Requirements with lower information content are superordinate (*generalises*) or with higher information content are subordinate (*refines*). If no differences in the information content occur (*similar*), then these requirements have to be critically examined in the further development process and changed if necessary. For support in decision problems, relations of the type *Relationship* are essential. With this type, a relationship of requirements can be considered based on their characteristics. It can be modeled, whether two requirements strengthen or weaken (*complementary*) each other or whether they are contrary to each other (*contrary*). These types of relations are extended by a quantitative specification to model the *intensity* and the *direction* (positive, negative) of these relations. The quantitative specification is for example based on mathematical relations between requirements or a conversion from qualitative relation value to a quantitative reference value. In addition, the types *Cost*, *Value* and *Satisfaction* exist, but these will not be considered further, since no cost nor value driven approach is to be used. In addition, traceability is not focused, since primarily the

https://doi.org/10.1017/pds.2021.505 Published online by Cambridge University Press

necessary relations to support decision making are addressed. Nevertheless, the approach was developed in a way that further relations can be integrated and thus aspects regarding cost and value as well as traceability can be considered. A summary of the relation types of the requirements and evaluation criteria is shown in Figure 2.

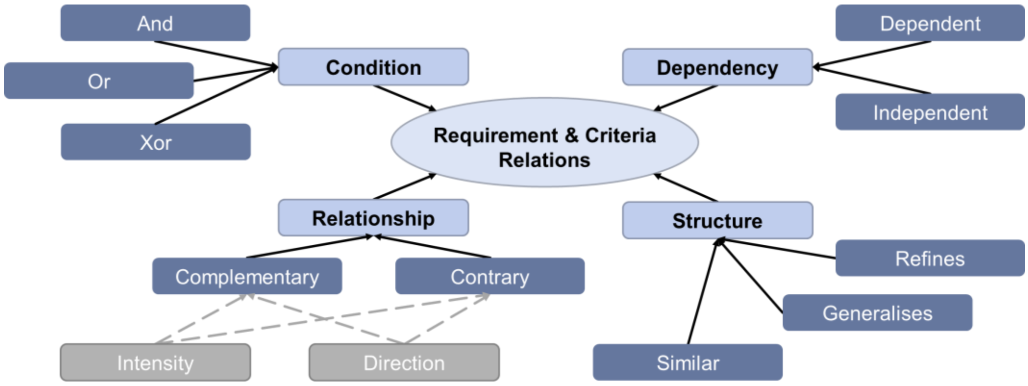

Figure 2. Types of requirement and criteria relations derived from literature

4.3 Relational modeling of requirements and evaluation criteria

At the beginning and during development, different requirements are defined, which are (semi-) automated derived into evaluation criteria according to the approach presented by Horber *et al.* (2020). This approach is therefore used as a starting point for the developed approach to first set up the relational model for requirements and then use the knowledge from the (semi-)automated derivation in decision making. The developed approach follows the procedure shown in Figure 3.

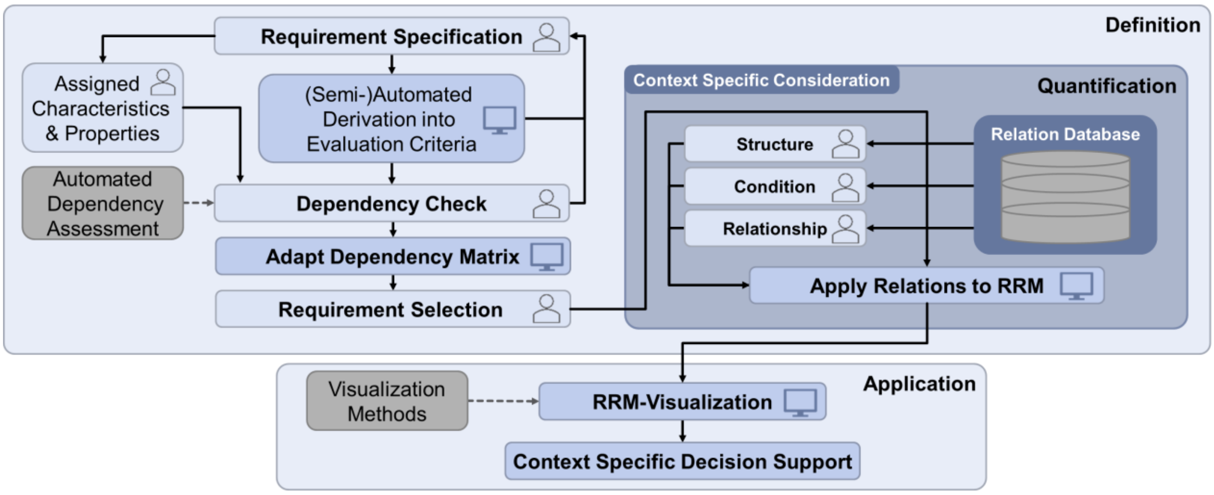

Figure 3. Approach for the modeling of relations between requirements and their related evaluation criteria in order to support decision-problems

Starting point for the procedure is the derivation of the evaluation criteria based on all requirements. More requirements are gradually added to the model. In iterative steps, the product developer must then determine the potential relations between the requirements in terms of dependencies. The determination takes place on the basis of assigned characteristics and properties of the requirements, therefore developers with appropriate knowledge and experience are necessary. Requirements are assigned to characteristics or with a 1:1 relation to properties. In a hierarchical manner, single or multiple characteristics are linked with a superordinate property, which will be used in decision-making as a metric for the determination of the achievement of an assigned development goal. Linked characteristics and properties are then systematically used to identify dependencies between derived criteria. From a certain degree of system complexity on, it is recommended to support this process by an automated identification of relations. For example, dependencies can be clustered based on the hierarchical structure of the requirements and dependencies can be derived. In addition, the semantic description of the requirement statement can provide information about the relevant subsystem or

https://doi.org/10.1017/pds.2021.505 Published online by Cambridge University Press

system element. This information can be captured with the help of natural language processing and used for dependency analysis. Further possibilities for a supported identification of dependencies can be found in (Robinson *et al.*, 2003) or (Graeßler *et al.*, 2020). In the following steps of the proposed procedure, context-specific requirements for a tangible decision problem are identified by the user and the necessary relations regarding relationship, structure and condition are determined. In order to use these relations in decision making, they have to be transferred into a suitable model. In this paper we therefore use matrix-based modeling. An advantage of this method is the systematic modeling of a complex system (Browning, 2016). For this reason, one submatrix per relation type is used to model the relations (see Figure 4).

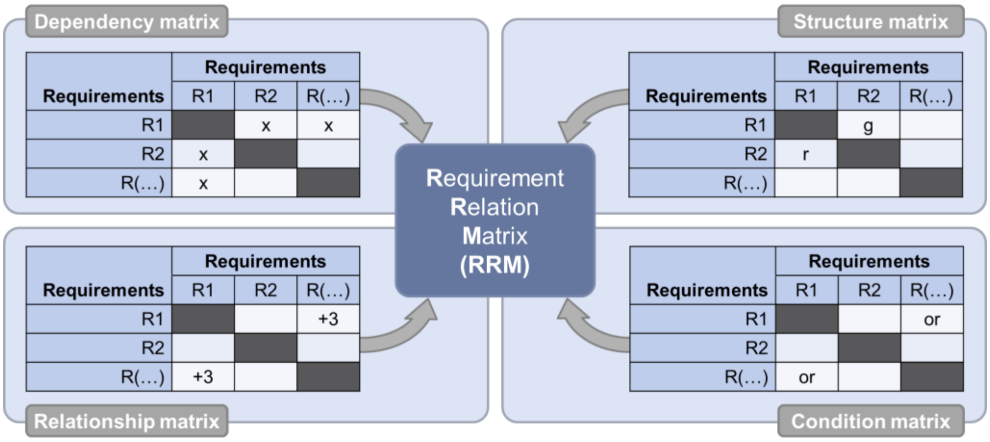

Figure 4. Types of requirement relation submatrices combined to a Requirement Relation Matrix (RRM)

The defined relations are stored in a database and serve as a knowledge source for future projects. This database can be set up with already completed projects, where an analysis of the requirement relations according to the proposed approach is done afterwards in order to support new projects. Nevertheless, a certain level of experience and knowledge of the developer is necessary for the requirement relation definition. Finally, the submatrices are summarized in the RRM and presented for the context-specific decision problem using suitable visualization methods. A common method is the graph-based representation of matrix-based models (Browning, 2016), which is also used here. With the help of the graphical representation, clusters in the system can be quickly captured and then analyzed context-specifically. The analysis provides information about the relations in the system as well as the derived criteria regarding the decision problem.

4.4 Application

The presented approach was applied to an exemplary decision situation where a test laboratory has to be extended by a test bench for coating testing. This is a purchase decision under consideration of 81 different requirements, where three different alternatives are available. The aim is to identify the test bench that meets the given requirements as best as possible. In this context, a software demonstrator was developed, which supports the specification of requirements as well as their. The demonstrator enables the comparison of two requirements in a pairwise manner and includes two functions to simplify the determination of relations. The dependency mode is primarily used to identify the dependencies (dependent or independent) between requirements. In the next step, a second mode can be used to display only those pairs of requirements that have a dependency. Then the relations regarding condition, structure and relationship are determined and saved into the database, which contains the requirements, their relations as well as the information about the criteria derivation. In the background the submatrices are filled automatically and the RRM is generated. The generated RRM can then be converted into a graph-based model with the help of an export function and used for system analysis (see Figure 5).

The graph-based visualization is done with the open source software Neo4j and the graph query language cypher. The visualization consists of requirements as nodes and the modeled relations as edges. All necessary information, e.g. requirement IDs, relation types or requirement specifications are stored in the model and can be displayed if selected by users. Furthermore, the results of the (semi-)

https://doi.org/10.1017/pds.2021.505 Published online by Cambridge University Press

automated derivation into evaluation criteria are linked with the relevant requirements. In the visualization window, different clusters can be identified and the requirements as well as their relations can be analyzed context-specifically. Figure 5 b) shows the exemplary cluster for time-resolved data acquisition (ID29). The cluster shows the requirements for noise measurement, which are linked to the superordinate requirement via a refine relation. For the decision, which test bench is purchased, it is not of interest whether the noise measurement is performed on one test disc (ID61) or on two test discs (ID62). However, all test benches must allow noise measurement in a frequency range of less than 10 kHz (ID64) and more than 1 MHz (ID65). Because the given alternatives are customer-specific, the knowledge generated from this could then be used to improve the alternatives by exchanging information with the manufacturer about possible extensions of the test stand.

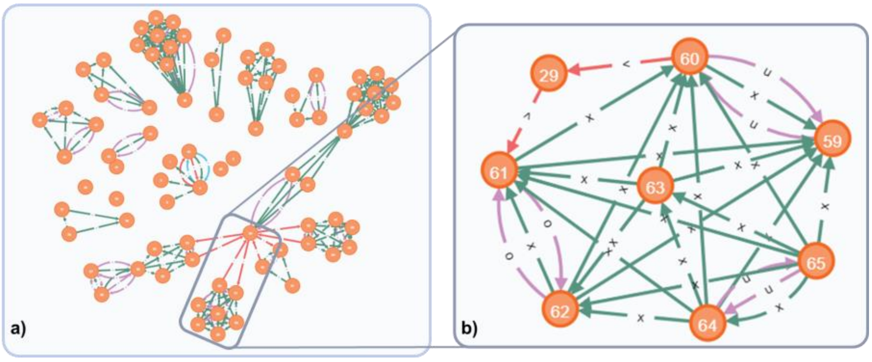

Figure 5. Graph-based visualization of requirement relation matrix for test bench purchase, a) reduced relational model and b) extraction of an exemplary cluster (rearranged)

The identified relations between requirements based on characteristics and properties can then be used in value function based decision-making (see Figure 6). In a weighted sum approach with dependent criteria, the relations have to be considered when setting up additive or non-additive functions in order to find the best solution. Users of the support tool receive information about the necessary relations from the matrix and graph-based visualization. Where it is more intuitive for users to locate clusters in a graph, the relations can then be extracted from the model. This enables users to handle requirement interactions and dependencies in multi-criteria decision-making.

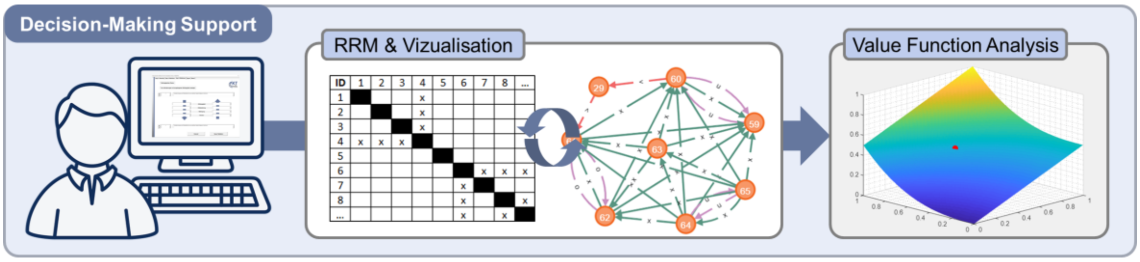

Figure 6. Analysis of dependent value functions based on graph-based visualization

5 DISCUSSION

In this contribution an approach is presented, with supports product developers regarding relational modeling of requirements in complex systems development and providing this knowledge to decision making. The relations between the requirements are already determined and modeled during the requirements specification and are therefore already accessible for developers in early phases. In conjunction with the (semi-)automated derivation into evaluation criteria it is possible to transfer this relational modeling of the requirements model to the criteria model. Relations between evaluation criteria can now be taken into account already in early stages of development. The matrix-based modeling allows a structured analysis of the system and the graph-based visualization is a suitable way to understand system structures regarding requirements and criteria models.
The presented approach takes up the necessity and importance of an continuous requirements management as described in Pahl *et al.* (2021) and allows the consistent transfer of requirements into the

https://doi.org/10.1017/pds.2021.505 Published online by Cambridge University Press

decision-making process. In contrast to the common methods for matrix-based modelling of relations, in particular Axiomatic Design (Suh, 1990) and House of Quality (Sullivan, 1986), it is possible with the presented approach to model relations consisting of different types (*condition*, *relationship*, *structure*) and not only on the basis of mathematical relations (Axiomatic Design) or qualitative relations (House of Quality). Especially the relations of the type condition as well as logical relations are not considered in those approaches. In addition, the approach presented is designed in such a way that the relations are only modelled context-specifically in detail, which reduces the effort required for the initial development of the model. Furthermore, coupled systems, in terms of dependent requirements, can be modelled e.g. through relations of the types *structure* or *relationship*. Nevertheless, it is possible to integrate the relations identified with Axiomatic Design or the House of Quality into the presented approach. Both approaches can be covered with the proposed relation type "relationship" (along with the intensity and direction). However, this must be examined in detail within future work.
However, the development of complex systems involves a great number of requirements as well as complex interactions. Although these interactions can be modeled and analyzed effectively and systematically, manual effort is still necessary. A significant reduction of the effort is achievable through an automated identification of dependencies between requirements. However, the presented concepts for automation still need further development and have to be evaluated. However, it is possible to integrate existing approaches into the presented procedure (a selection of approaches can be found in (Graeßler *et al.*, 2020)). Additionally, it must be examined, which effects occur when changes are made to the relational model and how these can be taken into account in order to maintain consistency within the models. The practical application with an example shows the applicability of the concept, but also confirms the impressions regarding the modeling effort. The purchase decision could be supported purposefully with the present approach. A conceptual demonstrator was developed in order to support the applicability, but further development is necessary as well as a user test.

6 CONCLUSION AND OUTLOOK

The present contribution emphasizes the lack of a procedure for decision making based on all requirements as well as in consideration of the relations between requirements in complex product development. The derivation of requirements into evaluation criteria is the basis for requirements-oriented decision-making. However, the lack of knowledge about the interactions between the requirements complicates the decision making process and an unused potential of the modeling remains. Based on this, this contribution first deals with the question of how requirements and the criteria derived from them can be related to each other in a structured way and how these relations can be modeled. For this purpose, the relation types in the literature are identified and reduced to the relations necessary for the presented approach. The modeling is matrix-based and provides a submatrix for each relation type, which can finally be combined to a requirement relation matrix (RRM). Furthermore, the question is answered, how a modeling procedure looks like and how the results can be used in the decision making process. The procedure includes the (semi-)automated derivation of requirements into criteria and uses the relations modeled in the RRM in order to transfer them to the criteria model. The results can then be analyzed context-specifically in the matrix-based or graph-based visualization and used for decision making. The applicability of the concept was tested and confirmed by a purchase example. A software demonstrator was developed to support the applicability. The relations between requirements can now be modeled and transferred to the criteria model. However, a manual specification effort regard relation modeling remains, which can be reduced by integration of existing approaches for automated relation identification. However, this is an opportunity for future research topics in this area. Furthermore, relational modeling is only possible in the matrix-based representation or in the software demonstrator so far. A bidirectional creation of relations in the graph-visualization would to improve the usability of the concept further.

ACKNOWLEDGMENTS

The authors thank the German Research Foundation (DFG) for supporting the research project 'Multi-criteria evaluation in product development [MuBePro] – Computer-supported evaluation process of solution variants with regard to all requirements from the product life cycle' under the grant number WA 2913/33-1.

https://doi.org/10.1017/pds.2021.505 Published online by Cambridge University Press

REFERENCES

Breiing, A. and Knosala, R. (1997), Bewerten technischer Systeme: Theoretische und methodische Grundlagen bewertungstechnischer Entscheidungshilfen, Springer Verlag, Berlin. http://dx.doi.org/10.1007/978-3-642-59229-4.

Browning, T.R. (2016), "Design Structure Matrix Extensions and Innovations: A Survey and New Opportunities", IEEE Transactions on Engineering Management, Vol. 63 No. 1, pp. 27–52. http://dx.doi.org/10.1109/TEM.2015.2491283.

Carlshamre, P., Sandahl, K., Lindvall, M., Regnell, B. and Natt och Dag, J. (2001), "An industrial survey of requirements interdependencies in software product release planning". Fifth IEEE International Symposium on Requirements Engineering, Toronto/Canada, 27.-31.08.2001, IEEE Computer Society, Los Alamitos, pp. 84–91. http://dx.doi.org/10.1109/ISRE.2001.948547.

Carlsson, C. and Fullér, R. (1995), "Multiple criteria decision making: the case for interdependence", Computers & Operations Research, Vol. 22 No. 3, pp. 251–260. http://dx.doi.org/10.1016/0305-0548(94)E0023-Z.

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, Vol. 31, Springer, Berlin, pp. 95–116. http://dx.doi.org/10.1007/3-540-28244-0_5.

Dermeval, D., Vilela, J., Bittencourt, I.I., Castro, J., Isotani, S., Brito, P. and Silva, A. (2016), "Applications of ontologies in requirements engineering: a systematic review of the literature", Requirements Engineering, Vol. 21 No. 4, pp. 405–437. http://dx.doi.org/10.1007/s00766-015-0222-6.

Eben, K.G.M. and Lindemann, U. (2010), "Structural Analysis of Requirements. Interpretation of Structural Criterions". , Cambridge/UK, 22.-23.07.2010, Hanser Verlag, München, pp. 249–261.

Graeßler, I., Preuß, D. and Oleff, C. (2020), "Automated Identification and Characterization of Requirement Dependencies - Literature Study to Compare Solution Approaches". Proceedings of the 31st Symposium Design for X, 16-17 December 2020, The Design Society, pp. 199–208. http://dx.doi.org/10.35199/dfx2020.21.

Heim, P., Lohmann, S., Lauenroth, K. and Ziegler, J. (2008), "Graph-based Visualization of Requirements Relationships". Requirements Engineering Visualization, Barcelona/Spain, 08.09.2008, IEEE CS Digital Library, Online, pp. 51–55. http://dx.doi.org/10.1109/REV.2008.2.

Horber, D., Schleich, B. and Wartzack, S. (2020), "Conceptual model for (semi-) automated derivation of evaluation criteria in requirements modelling". DESIGN2020, Online-Conference, 26.-29.10.2020, Cambridge University Press, pp. 937–946. http://dx.doi.org/10.1017/dsd.2020.52.

Kamata, M.I. and Tamai, T. (2007), "How Does Requirements Quality Relate to Project Success or Failure?". RE 2007, New Delhi/India, 15.-19.10.2007, IEEE Computer Society, Los Alamitos/USA, pp. 69–78. http://dx.doi.org/10.1109/RE.2007.31.

Karlsson, J., Olsson, S. and Ryan, K. (1997), "Improved practical support for large-scale requirements prioritising", Requirements Engineering, Vol. 2 No. 1, pp. 51–60. http://dx.doi.org/10.1007/BF02802897.

Morkos, B., Shankar, P. and Summers, J.D. (2012), "Predicting requirement change propagation, using higher order design structure matrices: an industry case study", Journal of Engineering Design, Vol. 23 No. 12, pp. 905–926. http://dx.doi.org/10.1080/09544828.2012.662273.

Öztürk, Z.K. (2006), "A review of multi criteria decision making with dependency between criteria". MCDM 2006, Chania, Greece, 19.-23.06.2006, Springer.

Pahl, G., Beitz, W., Bender, B. and Gericke, K. (2021), Pahl/Beitz Konstruktionslehre: Methoden und Anwendung erfolgreicher Produktentwicklung, 9. Auflage, Springer Vieweg, Berlin.

Ramesh, B. and Jarke, M. (2001), "Toward reference models for requirements traceability", IEEE Transactions on Software Engineering, Vol. 27 No. 1, pp. 58–93. http://dx.doi.org/10.1109/32.895989.

Robinson, W.N., Pawlowski, S.D. and Volkov, V. (2003), "Requirements interaction management", ACM Computing Surveys, Vol. 35 No. 2, pp. 132–190. http://dx.doi.org/10.1145/857076.857079.

Robinson, W.N. and Volkov, S. (2007), Conflict-Oriented Requirements Restructuring.

Rupp, C., Simon, M. and Hocker, F. (2009), "Requirements Engineering und Management", HMD Praxis der Wirtschaftsinformatik, Vol. 46 No. 3, pp. 94–103. http://dx.doi.org/10.1007/BF03340367.

Saaty, T.L. (2004), "Decision making — the Analytic Hierarchy and Network Processes (AHP/ANP)", Journal of Systems Science and Systems Engineering, Vol. 13 No. 1, pp. 1–35. http://dx.doi.org/10.1007/s11518-006-0151-5.

Sipahi, S. and Timor, M. (2010), "The analytic hierarchy process and analytic network process: an overview of applications", Management Decision, Vol. 48 No. 5, pp. 775–808. http://dx.doi.org/10.1108/00251741011043920.

Soomro, S., Hafeez, A., Shaikh, A. and Musavi, S.H.A. (2014), "Ontology Based Requirement Interdependency Representation and Visualization". IMTIC 2013, Jamshoro/Pakistan, 18.-20.12.2013, Springer Verlag, Cham, pp. 259–270. http://dx.doi.org/10.1007/978-3-319-10987-9.

Suh, N.P. (1990), The principles of design, Oxford University Press, New York.

Sullivan, L.P. (1986), "Quality function deployment", Quality Progress, Vol. 19 No. 6, pp. 39–50.

https://doi.org/10.1017/pds.2021.505 Published online by Cambridge University Press