

Designing Dynamic Decision Support for Electronic Requirements Negotiations

Annika Lenz¹

¹ University of Hohenheim, Department of Information Systems I, Stuttgart, Germany
annika.lenz@uni-hohenheim.de

Abstract. Decision support in software development is particularly important for requirements negotiations to help assessing requirements and their different implementation alternatives. Changes related to requirements are likely, which impede decision making. To keep an overview of the assessment of requirements and to keep this assessment up-to-date throughout a software development project, flexible decision support processes are needed. In this paper, we design interactive dynamic decision support, which can handle changes related to requirements dynamically. The designed support component is compared to two state-of-the-art approaches for decision support in requirements negotiations.

Keywords: Dynamic decision support, requirements negotiation, dynamic preference measurement

1 Introduction

To keep an overview of the various requirements in large software development projects is a great challenge, even if requirements management systems are employed. It gets even more challenging, if negotiations must be conducted, to determine, which requirements are actually implemented and how. Requirements negotiation is an “iterative process of communication and decision-making between [stakeholders] who have the overall goal of agreeing on a software development process and outcome” [1, p. 304]. In such coordination and reconciliation processes, requirements to be implemented, their development cost, and the delivery schedule are negotiated [2].

Decision support in such scenarios enables the quantification of requirements and possible alternatives of their implementations and thus makes them comparable [3]. In negotiations, in which various stakeholders (e.g. customers, developers, project managers, product owners) have a say, knowing one’s own position and expressing this position quantitatively provides enormous benefits to enforce one’s own interests.

A drawback is that software development is an industry characterised by a high degree of dynamics [4]. Especially in large software development projects, new information regarding requirements is obtained or disclosed throughout the project, so changes in requirements are likely [5]. An existing set of requirements must be refined in subsequent iterations of the requirements negotiation process in terms of additional requirements, omitted requirements, and/or changes in existing requirements’ scope [2,

6], which illustrates the high degree of uncertainty and incompleteness of information [7].

Moreover, negotiations underlie process dynamics which influence the negotiators' preferences [8, 9]. In general, preferences are unstable due to new preference formation, learning or fatigue [10]. This effect is amplified by the negotiation partners' exchange during the negotiation process. By the negotiation partners' reciprocal behaviour they gain new information or achieve clarity about the negotiation issues or values, which influences their preferences [11].

Thus, preference changes in requirements negotiations may stem from agenda changes [12] or from the reciprocity of the negotiation process itself. In both cases, existing measurement of preferences must be efficiently adjusted to provide meaningful decision support. Dynamic decision support takes the perspective of time into consideration. It considers an unstable negotiation agenda (i.e. requirements or their implementation solution changes), and unstable preferences themselves.

From a requirements engineering perspective, decision making in requirements negotiations is supported by multi-criteria decision analysis approaches [e.g. 13], which methodologically consider a dynamic perspective in their requirements negotiation process, however, do not carry it to the decision support process. Negotiation research delivers decision support dealing with incomplete information [e.g. 14]. However, literature, which integrates insights of both domains is still scarce [2, 15]. There is no comprehensive approach that focuses on supporting decisions in requirements negotiations on a dynamic, interactive level to allow for adjustment of preferences [cf. 15], although the necessity of dynamic methodologies for requirements negotiations is present [e.g. 6, 11].

Against this background, our aim is to design efficient dynamic decision support in requirement negotiations integrating a requirements engineering and negotiation research perspective. Dynamic decision support enables continuously accurate preferences, which allows an accurate negotiation analysis [16], i.e. analysis of own and the negotiation partner's requirements specification offers and concessions. To this end, the aim of our paper is two-fold: (1) to design a dynamic decision support component for requirements negotiations; (2) to evaluate its suitability by comparing it with state-of-the-art decision support in electronic requirements negotiations using a scenario-based approach [17, 18].

2 State-of-the-Art

In negotiations, the two perspectives of individual decision making and plural decision making are relevant [16]. Decision analysis takes an individual perspective and focuses on a normative and prescriptive approach for an individual negotiator or a single negotiation party, while negotiation theory as a perspective of plural decision making assumes joint decision making. Negotiation theory describes how real people could make better collaborative decisions. Both perspectives are integrated in asymmetric negotiation analysis, which supports a negotiator's position considering the partner's behaviour [14] to support achieving the best outcome [16]. Thus, decision support takes

an individual decision making perspective in a joint decision making context, whose result and benefit is asymmetric negotiation analysis.

Prerequisite is that the individual preferences of the negotiating parties are measured beforehand. A multitude of preference measurement methods exists to quantify the decision makers' preferences. As terminology differs in the field of preference measurement methods, we refer to measuring preferences for attributes and alternatives. In self-explicated approaches [19], the decision makers elicit their preferences directly. The preference measurement can thus be carried out quickly and easily and the cognitive complexity can be kept low for the decision maker. For example, Adaptive Self-Explication (ASE, [20]) comprises of three different user dialogue steps: (1) the alternatives are rated on a scale from 0 to 1; (2) the attributes are ranked regarding their importance; (3) pairwise comparisons of attributes are conducted, for which two attributes are presented to the decision maker, who enters a ratio of how much more important one attribute is compared to the other. Another popular preference measurement methods is TOPSIS [21], which uses linguistic terms to determine the attribute weightings.

Electronic negotiation systems aim to support the three phases of a negotiation, namely the planning phase (or pre-negotiation or preparation), the actual negotiation phase, and the post-settlement phase. Negotiation support systems have an inherent characteristic of providing decision support [22]. They help decision makers to understand the problem and assess the implications of their decisions. To do so, preference measurement methods are applied to calculate the utility of single offers using multi-attribute utility theory [23] based on the importance weightings of negotiation issues (= attributes) and issue values (= alternatives) [e.g. 24–26]. The resulting utility model forms the basis for negotiation analysis [16].

Requirements engineering distinguishes between requirements and their implementation solutions [17]. Using the terminology of preference measurement, requirements refer to attributes and solutions refer to alternatives. In requirements negotiations, three types of issues are negotiated: i) design related issues, ii) contract related issues, iii) technological issues and for each implementation solutions respectively options to resolve the issue [2, 27, 28]. For the sake of better reading, we refer to all types as requirements and their solutions.

Involved negotiation parties cover success critical stakeholders of the software development project [6], who typically pursue mismatching goals: Future end users desire numerous features, a high service level, or fast delivery, buyers may also wish timely delivery, however are also interested in cost within budget, or compliance, while developers prefer stable requirements and flexible contracts [2]. Typically, only a relevant subset of these stakeholders are actually involved in the respective negotiation, which results in different negotiation constellations [29]. Although requirements negotiation is an iterative process [2], which is not a one-time episode but an on-going process in general, its instantiation characteristics depend on several factors such as the software development method deployed (e.g. traditional, agile, or hybrid software development), the project type, the collaboration situation [2], involved stakeholders [29], the project stage (beginning or completion of the project), negotiation scope (i.e. requirements and solutions, technologies, or contract aspects), or frequency. Thus, their

structure may differ. Requirements negotiations can be performed in independent phases, each including respective stakeholders and resulting in a (partial) agreement [29].

Nonetheless, in almost every software development project, requirements negotiations are present in some way, and in each requirements negotiation, decisions must be made. Decision support accordingly is required for stakeholders' conflicting requirements perspectives in the actual negotiation phase. We consider negotiations, in which preference measurement is supported electronically and analysis is provided based thereon. Preference measurement can jointly be conducted in face-to-face negotiation workshops [e.g. 6] or be applied asynchronously for an individual decision maker to prepare the requirements negotiation [e.g. 30].

The most widely used methodology in the area of interactive requirements negotiation support is the WinWin methodology [6, 15], which aims to achieve a fair agreement among all involved stakeholders by attempting to meet the win conditions of each stakeholder [6]. Dealing with various stakeholders, group recommendation technologies, such as IntelliReq [31], dedicatedly aim to support collaboration and enable group decision making [15]. Based on the stakeholders' preferences, recommendation technologies are applied to reach a joint decision on which requirements to implement (first).

The presented existing approaches follow a dynamic paradigm and provide decision support, however dedicated preference adjustment mechanisms are missing. Applying dynamic preference measurement to the domain of requirements negotiations can improve such approaches substantially. Our work addresses the depicted gap by designing and evaluating such a concept.

3 Methodology

The methodology applied in this paper covers the design of dynamic decision support for requirements negotiations, DynaDeS, and the evaluation of the designed component, both based on scenarios to be captured [17, 18, 32].

We design the component to fulfil the goal of providing dynamic decision support to handle preference changes efficiently. To do so, scenarios are considered, for which the design goal must be fulfilled [17, 18]. The scenarios affecting preference information changes, which must be supported, stem from volatile requirements [2] and negotiation process dynamics [8, 11], see Table 1. When requirements or solutions change during the negotiation process, the decision support component is intended to provide a process to adjust their preferences accordingly. Thus, it must consider (A) a new requirement with new solutions, (B) a new solution without a new requirement and adjust their preferences. Moreover, preferences are unstable and may change over time despite of requirements or solution changes, thus the decision support component must handle (C) preference changes for requirements and (D) preference changes for solutions.

To reach the overall aim of handling preference changes efficiently, the design process conducts the following steps: (1) Existing decision support of negotiation

research is adapted to requirements negotiations; (2) the preference measurement is extended to allow dynamic preference adjustment; (3) the decision support processes are adapted to meaningfully use the adjusted preferences.

Table 1. Scope of preference adjustment for dynamic decision support in requirements negotiations

#	Scenario	Cause
A	New requirement	Unstable requirements
B	New solution	Unstable requirements
C	Changed requirement preference	Negotiation process dynamics
D	Changed solution preference	Negotiation process dynamics

The designed DynaDeS is implemented in the web-based negotiation support system Negoisst [26, 30] to exploit the benefits of an integrated system. DynaDeS is evaluated by means of a scenario-based [17, 32] comparison to analyse whether the designed dynamic decision support component meets its design goals for requirements negotiations. Scenarios illustrate how design goals are satisfied [17]. Thus, we compare DynaDeS to existing decision support w.r.t. to their suitability for the required scenarios (cf. Table 1). Furthermore, as the WinWin methodology is the methodology most commonly used for electronic requirements negotiations [15], we include decision support within the WinWin methodology [6, 33]. Moreover, we compare IntelliReq [31], which provides dedicated group decision support in requirements negotiations. Although the support components to compare are heterogeneous in different characteristics, they coincide in taking a dynamic perspective on decision support for requirements negotiations.

4 Design of the Dynamic Decision Support Component

The decision support component is not intended to replace requirements elicitation processes or requirements refinement processes. It starts when an initial set of requirements and their implementation solutions is determined. This set may be rudimentary at the beginning and be refined during the superordinate negotiation process, however, requirements and solution refinement is not meant to be part of the decision support component. In contrast, the decision support component is supposed to handle the required scenarios of changes of the environment.

4.1 A Concept for Dynamic Decision Support

In the following, we will (1) adapt existing decision support of negotiation research to requirements negotiations; (2) enable dynamic preference adjustment; (3) design processes to meaningfully use the adjusted preferences.

(1) For decision support during requirements negotiations without a dynamic perspective, information about the agenda (i.e. requirements and solutions to negotiate) is gathered in the planning phase. The problem definition takes place based on the

project goals and the project phase [cf. 34]. At a very early stage, requirements negotiation involves negotiation issues on a high-level, while negotiation in a later stage of the software development project focuses on specific aspects or sub-projects. Requirements that are elicited at an early stage allow for a wide range of implementation solutions that will be specified later. Furthermore, in the planning phase success critical stakeholders' individual goals are collected. These goals may concern high-level issues, general system functions, budget, schedule, or technical issues, such as the development environment. Having extracted the initial agenda to negotiate, the negotiation issues and values are elicited. Subsequently, still during the planning phase, an initial preference elicitation is performed, resulting in a (tentative) utility model, which is required to provide asymmetric analysis in the actual negotiation phase.

During the actual negotiation phase, the agreement takes place. The stakeholders exchange offers and counteroffers to find a mutually beneficial and acceptable solution. Analytical support is provided to assess own offers or offer drafts as well as the negotiation partner's offers based on the preferences measured.

(2) To provide dynamic preference adjustment, we choose ASE [20] as basis preference measurement method, since it performs very well regarding its efficiency, validity, and cognitive complexity [35]. Furthermore, although it is a one-shot preference measurement method as is common in multi-criteria decision analysis, it provides the potential to be adopted for dynamic contexts [35], since it uses scarcely dependencies between preference information and allows to separate preference information measured and a utility model calculated based thereon. In this paper, we incorporate DynASE, which extends ASE by dynamic preference mechanisms, into DynaDeS.

For the dynamic preference adjustment, similar user dialogues as in the initial preference elicitation are applied, namely rating of the desirability of all solutions, ranking of the requirements according to their importance, and pairwise comparison of all requirements [20]. After a bundle of pairwise comparisons is conducted, a utility model can be calculated. An adaptive selection of pairwise comparisons facilitates to ask for the greatest possible information gain. If the resulting utility model is sufficiently precise, the preference adjustment process terminates. Otherwise, further pairwise comparisons are conducted until a pre-defined number of comparisons is reached, or all issues are compared at least once.

(3) The decision support process is adapted to react to negotiation process dynamics and to embed subsequent activities in the actual negotiation phase. It allows to adjust preferences and to re-calculate the utility model during the actual negotiation phase and shifts thereby activities of the planning phase into the negotiation phase (similar to agenda negotiations, cf. [12]), see Figure 1. Focus of the dynamic preference adjustment is the interaction of preference measurement and the utility model calculation. Most preference measurement methods do not separate the pure measurement of preferences from the calculation of the utility model. However, to enable efficient dynamic decision support, the possibility of separation is inevitable. The preferences resulting from the initial preference elicitation must be saved

independently of the utility model calculated to reuse still valid preferences in the case of changes.



Figure 1. Dynamic decision support in the negotiation phases

In DynaDeS, the dynamic aspect comes into effect if new information is obtained, which result in (A) requirements, (B) solutions, (C) requirement preference, and/or (D) solution preference changes, see Figure 2. The planning phase is entered again in all scenarios. In the scenarios of A and B, additionally the agenda is adjusted. Subsequently in all scenarios, preferences are adjusted as well, dependent on the information gained: Solution ratings are maintained from the existing preference information if it is still valid. This can be facilitated because the solution ratings of one requirement do not relate to the ratings of other solutions. Except for scenario C when only requirement preference information changes, the first step of the preference measurement is repeated for (A) solutions of the new requirement, (B) a new solution, or (D) the outdated solution preference information. In the scenarios B and D where no requirement information has changed, the user dialogue is already completed and a renewed accurate utility model is calculated based on existing valid preference information and newly gathered preference information, which can be used in the stakeholder's analysis. For the scenarios of A and C, the preference adjustment process continues, presenting the existing requirements ranking to the user, in which (A) the new requirement must be sorted respectively (C) the rank of the respective requirement must be corrected. Preference information of paired comparisons of all other requirements is reused. Thus, based on the ranking and valid paired comparisons, a renewed utility model is calculated again. If the utility model is not yet sufficiently precise, additional pairwise comparisons involving the respective requirement can be performed. Optionally, an interactive check allows the stakeholders to view and validate their utility model and initiate preference adjustment in case the utility model does not fit their preferences (anymore). Again, based on a newly calculated utility model, precise analytical support can be provided.

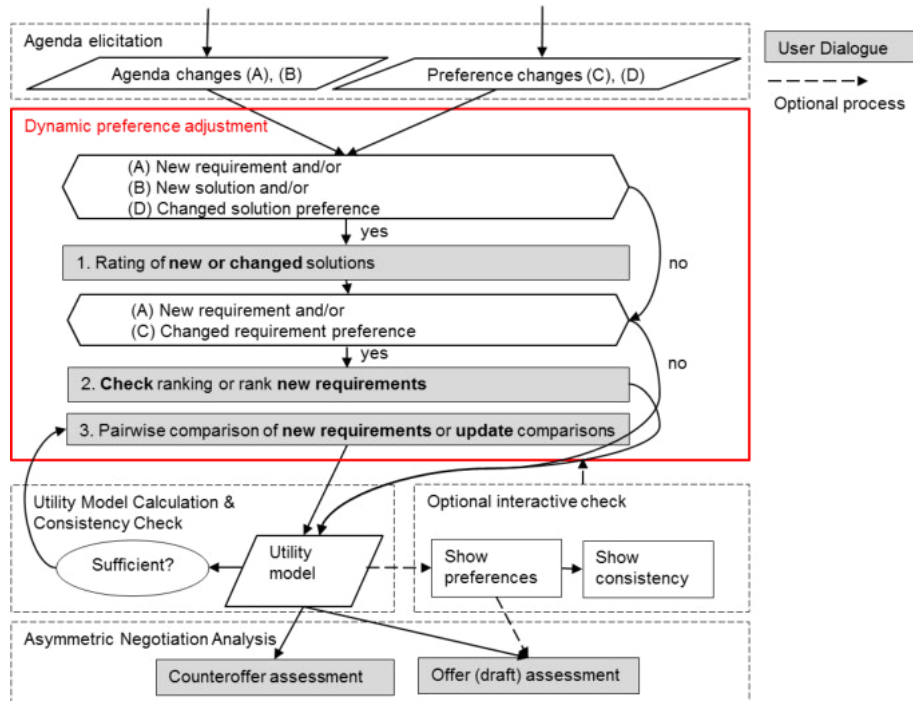


Figure 2. DynaDeS

4.2 Instantiation of Dynamic Decision Support in a Negotiation Support System

The designed dynamic decision support is implemented in the negotiation support system Negoisst [26, 30] as it already offers holistic negotiation support. Negoisst is a web-based support system, which provides decision support, communication support, and document management. One aspect of its communication-orientation is to support the offer exchange by semi-structured message exchange, which allows to send unstructured text messages along with a structured, formal offer. To make binding offers, the negotiators send each other semi-structured messages in an alternating order.

Negoisst provides extensive analytical features for decision support [e.g. 14]. The analytical features enable to assess offers quantitatively based on the negotiators' preferences by utility values. The utility of offers can be analysed by each negotiator. The history graph serves as visual support of the negotiators' convergence during the negotiation process, depicting the utility of offers over time. Concessions and gains can be analysed visually. Negoisst provides analytical support during the phases of planning (e.g. preference elicitation) and in the phase of negotiation (e.g. offer assessment and analysis). The implementation of the preference adjustment within DynaDeS covers the following three user dialogues: (1) rating of solutions; (2) ranking of requirements; (3) comparing requirements pairwise by using a slider or input fields, see Figure 3.

Please rate all values of the issue Registration

Please select your best and worst case.

Best Case: University accounts and guest accounts

Worst Case: ITL accounts

Please rate how desirable each of the following values are for you. Assume that all other issues remain the same.

University accounts: ○○○○○○●○○○○

University accounts and guest accounts: ●●●●●●●●●●●●●●●●

ITL accounts: ●●●●●●●●●●●●●●●●

Please rank the following improvements of issues by dragging and dropping them in the right order.

Ranking an issue does not imply a change in any best case, worst case, or other value.

Most important	Worst case → Best Case
≡ Standalone or integration	Integration → Standalone
≡ Registration	ITL accounts → University accounts and guest accounts
≡ Product name	ITL → GeSCo
≡ Interfaces	Manual slides import and data export → Manual slides import
≡ Brainstorming	No transformation → Transformation into survey
≡ Concurrent users	4 000 Users → 2 500 Users
Least important	Worst case → Best Case

Which of the improvements from worst case to best case represents a higher utility for you? The position of the slider indicates how important the issues are for you.

Standalone or integration	Concurrent users
Best Case: Standalone	Best Case: 2 500 Users
Worst Case: Integration	Worst Case: 4 000 Users

Standalone or integration is 3.00 times more important than Concurrent users

Ratio: 75 : 25

Figure 3. Preference adjustment in Negoisst: Rating of solutions, ranking of requirements, and pairwise comparison of requirements.

5 Scenario-Based Comparison of Dynamic Decision Support for Requirements Negotiation

In the following, we will analyse in a scenario-based [17, 32] comparison as to whether DynaDeS meets its design goals for requirements negotiations. DynaDeS will be compared with decision support based on the WinWin methodology [33] and IntelliReq [31].

5.1 Characteristics of the Decision Support Components to Analyse

The three decision support components all quantify the assessment of requirements and/or solutions. However, they use different assessment bases to provide decision support. In a late incorporation of the WinWin methodology, subsequently called WinWin, TOPSIS is used, which assesses strategic business goals and requirements

based on ease of realisation, their business value, and relative penalty [33], while IntelliReq requests the user to rate one of the solutions for a requirement as the preferred solution [31]. Requirements are not assessed. In contrast, DynaDeS uses utility functions to assess requirements and their solutions. It assesses the requirements relatively dependent on all requirements, while the solutions are assessed independently determining one most preferred, one least preferred solution and rating all other solutions directly on a scale from 0 to 1 per requirement.

Regarding the actual support for the negotiators to make their decisions, WinWin focuses on making well-informed decisions based on cost, benefit, and penalty. It additionally considers the requirements' contribution to business goals and performs a relative assessment, so requirements and their solutions can be compared with each other. IntelliReq provides a recommendation for the group of stakeholders, which is based on the preferences of all group members. The stakeholders' assessments do not relate to each other, nor do the assessment of the requirements relate to each other. DynaDeS assesses solutions for requirements under consideration in terms of utility based on the solution preferences and the relative importance of a requirement to facilitate utility analysis [16, 36].

The process for requirements negotiations using the WinWin methodology follows an iterative spiral model. Adjustments can be conducted in each iteration and decision support is correspondingly provided in each iteration when the win conditions are reconciled [6]. IntelliReq is intended to be used in any existing requirements negotiation process model. Since it is not embedded in a process, stakeholders can change their preferences any time as long as the process has not completed resulting in a group recommendation. DynaDeS allows an iterative refinement throughout the negotiation process. Preference adjustments can be conducted at any time to provide updated accurate decision support when asymmetric analysis in the negotiation process is needed.

The concept of requirements and solutions is viewed differently in the three decision support components. The WinWin methodology elicits and negotiates win conditions, issues, and options. To consolidate terminology, we refer to issues as requirements and options as solutions. IntelliReq uses requirements and decision alternatives. Here, we refer to decision alternatives as solutions, too. DynaDeS emerges from general electronic negotiation support. Here, negotiation issues are referred to as requirements and issue values are referred to as solutions.

5.2 Comparison of the Decision Support Components

In the following (see Table 2), we analyse WinWin, IntelliReq, and DynaDeS based on the scenarios from Table 1. Following the WinWin extensions of the spiral model [6], for scenario A, in each iteration a new requirement can be added. The subsequent steps need to be performed, which are to generate solutions to cover the requirement and to jointly agree upon both the new requirement and its solutions. The new requirement has to be assigned to business goals to correctly assess its value. The preference measurement must be repeated. A new solution for a requirement (B) can also be added in each iteration. The solution possibly makes agreements obsolete, so in the same

iteration, agreements possibly need to be dissolved and a new agreement must be made involving the new solution. Scenario B does not impact decision support, since solutions are not assessed. The process does not foresee that a stakeholder's preference may change after a requirement is assessed as required in scenario C. A repetition of the assessment and calculation is required. If a stakeholder's preference for a solution (D) has changed, is not applicable in WinWin, since solution preference information is not elicited and included in decision support.

Table 2. Comparison of Decision Support Components for Requirements Negotiations

	<i>Scenario A</i>	<i>Scenario B</i>	<i>Scenario C</i>	<i>Scenario D</i>
WinWin [33]	Repeat preference measurement	No impact on decision support	Repeat assessment for the respective requirement	Not applicable
IntelliReq [31]	Repeat recommendation process	Repeat recommendation process	Not applicable	Repeat recommendation process
DynaDeS	Rate solutions of new requirement; insert requirement into ranking; optionally conduct two requirement comparisons; repeat calculation of utility model	Rate new solution; repeat calculation of utility model	Check requirements ranking; conduct two requirement comparisons; repeat calculation of utility model	Assess solution; repeat calculation of utility model

In IntelliReq, adding a new requirement (A) is not designed. However, adding requirements to the agenda should be possible as long as the superordinate negotiation process is not concluded. For the new requirement solutions have to be entered and all stakeholders must utter their preferences for the best solution. The recommendation process must be repeated from scratch including the new requirement, so decision support can be provided. A new solution (B) is also not foreseen but should be possible to be added at any time. As in scenario A, all stakeholders must consider if the new solution is their most preferred solution and enter it accordingly. The recommendation process must be repeated again. Scenario C is not applicable since requirements are not assessed in IntelliReq. Thus, decision support regarding requirements is not provided. Stakeholders can add or change (D) preferences for solutions at any time. IntelliReq does not use a preference measurement method, but asks for the most preferred solution, while the remaining solutions are not assessed. Thus, changes can be made very easily and quickly. Subsequently the recommendation process must be repeated again.

In DynaDeS, in case of a new requirement (A), all solution preferences can be maintained. Only the solutions of the new requirement have to be assessed. The existing requirements ranking can be reused, too. The ranking is presented to the stakeholder

who only must sort in the new requirement. All paired comparisons can be kept. At this point, a utility model can already be calculated, because the relative importance weight of the new requirement can be interpolated based on the requirements ranking. The usual criteria terminate the adjustment process, i.e. two comparisons including the new requirement can be performed if desired. If a new solution is added (B) to an existing requirement, which does not replace the best or worst solution of the respective requirement, it is rated as in scenario A. The partial utility value of the newly rated solution is included in the utility model. In case the preference information for a requirement changes (C), the requirements ranking must be checked respectively corrected and all related paired comparisons must be adjusted. Based thereon, the utility model can be re-calculated. If the preference information of an existing solution has changed (D), the preference adjustment process is the same as in scenario B. The respective requirement and its solution ratings is presented to the stakeholder, who corrects the respective assessment.

6 Discussion and Conclusions

The scenario-based evaluation shows that although both decision support components besides DynaDeS follow dynamic approaches, they do not support preference adjustment reusing the existing assessment. It is required to repeat preference measurement instead of only adjusting changes. Consequentially, the applied preference measurement methods require repeated information input. This is the major advantage of DynaDeS: The analysed decision support component based on the WinWin methodology cannot deal with changes efficiently. Using DynaDeS, the correction of the requirement ranking is already sufficient to measure the relative importance of an issue based on valid relative importance weightings of other requirements. If a more accurate importance weighting is desired, two pairwise comparisons are sufficient.

The main contributions of this paper are the dynamic concept and the proposed decision support process to integrate it. The adjustment of preferences during the requirements negotiation process provides a more accurate basis for decision making that improves the effectiveness and efficiency of the requirements negotiation process. An increase in effectiveness is reached by incorporating new or changing information during the process. Key concepts are both the separation of preferences and utility and relating preferences only to one other object. The general concept can be used and adopted by existing approaches for decision support in requirements negotiations to extend their work. The scenario-based evaluation shows the general applicability of a dynamic perspective as an improvement to decision support. Implications for practice relate to the facilitation of quickly adjusting single preferences, representing the stakeholders' interests, and thus relying on accurate decision support.

Future research could address integrating dependencies between requirements and assumptions of multi-attribute utility theory, since it requires attributes must be independent of each other. Another extension to this work could address an additional level of criteria. Stakeholders may use various criteria to assess requirements or their

implementation solutions. For example, customers may evaluate requirements according to their importance, the budget, or the time of delivery, while developers use profitability and availability of resources. Dynamic decision support could enable individual criteria.

Moreover, research proposes that visualisation of information respectively the decision problem should both match the individual's cognitive style and the decision task characteristics [37]. The decision maker's experience could be considered as well [38]. Future studies could investigate, which type of visualisation in requirements negotiation is the most appropriate one. Different types of visualisation and information input could be provided so that an individual decision maker can pick the most appropriate one, e.g. regarding the presentation of the pairwise comparisons in the preference adjustment process. Additionally to different input types, the ratio of the comparison could be provided in numbers, in bars or in a bar chart.

Acknowledgements

The author gratefully acknowledges her PhD scholarship from the State Baden-Württemberg and the support of the research area "Negotiation Research – Transformation, Technology, Media, and Costs", Faculty of Business, Economics, and Social Sciences at the University of Hohenheim.

References

1. Lenz, A., Schoop, M., Herzwurm, G.: Requirements Analysis as a Negotiation Process. In: Kaminski, B., Kersten, G., Szufel, P., Jakubczyk, M., Wachowicz, T. (eds.) *The 15th International Conference on Group Decision & Negotiation letters*, pp. 303–309. Warsaw School of Economics Press (2015)
2. Grünbacher, P., Seyff, N.: Requirements Negotiation. In: Aurum, A., Wohlin, C. (eds.) *Engineering and Managing Software Requirements*, pp. 143–162. Springer Berlin Heidelberg (2005)
3. Vetschera, R.: Preference-Based Decision Support in Software Engineering. In: Biffel, S., Aurum, A., Boehm, B., Erdogmus, H., Grünbacher, P. (eds.) *Value based software engineering*. Springer, Berlin, Heidelberg (2006)
4. Jarke, M., Lytinen, K.: Editorial: "Complexity of Systems Evolution: Requirements Engineering Perspective". *ACM Trans. Manage. Inf. Syst.* 5, 1 (2015)
5. Hull, E., Jackson, K., Dick, J.: *Requirements engineering*. Springer, London (2011)
6. Boehm, B., Kitapci, H.: The WinWin Approach: Using a Requirements Negotiation Tool for Rationale Capture and Use. In: Dutoit, A.H., McCall, R., Mistrík, I., Paech, B. (eds.) *Rationale Management in Software Engineering*, pp. 173–190. Springer, Berlin, Heidelberg (2006)
7. Ngo-The, A., Ruhe, G.: Decision Support in Requirements Engineering. In: Aurum, A., Wohlin, C. (eds.) *Engineering and Managing Software Requirements*, pp. 267–286. Springer Berlin Heidelberg (2005)
8. Curhan, J.R., Neale, M.A., Ross, L.: Dynamic valuation. Preference changes in the context of face-to-face negotiation. *Journal of Experimental Social Psychology* 40, 142–151 (2004)

9. Vetschera, R.: Preference structures and negotiator behavior in electronic negotiations. *Decision Support Systems* 44, 135–146 (2007)
10. Netzer, O., Toubia, O., Bradlow, E.T., Dahan, E., Evgeniou, T., Feinberg, F.M., Feit, E.M., Hui, S.K., Johnson, J., Liechty, J.C., et al.: Beyond conjoint analysis. *Advances in preference measurement. Mark Lett* 19, 337–354 (2008)
11. Reiser, A., Krams, B., Schoop, M.: Requirements Negotiation in Consideration of Dynamics and Interactivity. In: *Proceedings of the REFSQ 2012 Workshops and the Conference Related Empirical Study, Empirical Fair and Doctoral Symposium, ICB Research Report No. 52*, pp. 163–174. Springer, Berlin (2012)
12. Fernandes, M.: *Agenda negotiations in electronic negotiation support systems. An information systems perspective. Books on Demand, Norderstedt* (2016)
13. Ruhe, G., Eberlein, A., Pfahl, D.: Quantitative WinWin – A New Method for Decision Support in Requirements Negotiation. In: *Proceedings of the 14th International Conference on Software Engineering and Knowledge Engineering*, pp. 159–166 (2002)
14. Reiser, A.: *Entscheidungsunterstützung in elektronischen Verhandlungen. Eine Analyse unter besonderer Berücksichtigung von unvollständigen Informationen. Springer Fachmedien Wiesbaden; Imprint: Springer Gabler, Wiesbaden* (2013)
15. Lenz, A., Schoop, M., Herzwurm, G.: Electronic Requirements Negotiation - A Literature Survey on the State-of-the-Art. In: *UK Academy for Information Systems Conference (ed.) Proceedings of UKAIS 2016, Oxford, paper 23* (2016)
16. Raiffa, H., Richardson, J., Metcalfe, D.: *Negotiation analysis. The science and art of collaborative decision making. Belknap Press of Harvard University Press, Cambridge, MA* (2002)
17. Pohl, K.: *Requirements engineering. Fundamentals, principles, and techniques. Springer, Heidelberg, New York* (2010)
18. Carroll, J.M.: *Making use. Scenario-based design of human-computer interactions. MIT Press, Cambridge, Mass* (2000)
19. Eckert, J., Schaaf, R.: Verfahren zur Präferenzmessung – Eine Übersicht und Beurteilung existierender und möglicher neuer Self-Explicated-Verfahren. *Journal für Betriebswirtschaft* 59, 31–56 (2009)
20. Netzer, O., Srinivasan, V.: Adaptive Self-Explication of Multiattribute Preferences. *Journal of Marketing Research* 48, 140–156 (2011)
21. Hwang, C.-L., Lai, Y.-J., Liu, T.-Y.: A new approach for multiple objective decision making. *Computers & Operations Research* 20, 889–899 (1993)
22. Arnott, D., Pervan, G.: A critical analysis of decision support systems research revisited. The rise of design science. *J Inf Technol* 29, 269–293 (2014)
23. Keeney, R.L., Raiffa, H.: *Decisions with multiple objectives. Preferences and value tradeoffs. Wiley, New York* (1976)
24. Kersten, G.E., Noronha, S.J.: WWW-based negotiation support. Design, implementation, and use. *Decision Support Systems* 25, 135–154 (1999)
25. Thiessen, E.M., Soberg, A.: SmartSettle Described with the Montreal Taxonomy. *Group Decis Negot* 12, 165–170 (2003)
26. Schoop, M., Jertila, A., List, T.: Negoisst: A Negotiation Support System for Electronic Business-to-Business Negotiations in E-Commerce. *Data & Knowledge Engineering* 47, 371–401 (2003)
27. Lenz, A., Schoop, M.: Decision Problems in Requirements Negotiations – Identifying the Underlying Structures. In: Schoop, M., Kilgour, D.M. (eds.) *Group Decision and Negotiation. A Socio-Technical Perspective: 17th International Conference, GDN 2017*,

- Stuttgart, Germany, August 14-18, 2017, Proceedings, pp. 120–131. Springer International Publishing, Cham (2017)
28. Herzwurm, G., Schoop, M., Reiser, A., Krams, B.: E-requirements negotiation: Electronic negotiations in the distributed software development. In: Mattfeld, D.C., Robra-Bissantz, S. (eds.) *Multikonferenz Wirtschaftsinformatik 2012. Tagungsband der MKWI 2012*, pp. 1859–1870. Gito; Univ.-Bibl, Berlin, Braunschweig (2012)
 29. Fricker, S., Grünbacher, P.: Negotiation Constellations - Method Selection Framework for Requirements Negotiation. In: Paech, B., Rolland, C. (eds.) *Proceedings Requirements Engineering: Foundation for Software Quality. 14th International Working Conference*. Springer, Berlin Heidelberg (2008)
 30. Schoop, M.: Support of Complex Electronic Negotiations. In: Kilgour, D.M., Eden, C. (eds.) *Handbook of Group Decision and Negotiation*, 4, pp. 409–423. Springer Netherlands (2010)
 31. Felfernig, A., Zehentner, C., Ninaus, G., Grabner, H., Maalej, W., Pagano, D., Weninger, L., Reinfrank, F.: Group Decision Support for Requirements Negotiation. In: Hutchison, D., Kanade, T., Kittler, J., Kleinberg, J.M., Mattern, F., Mitchell, J.C., Naor, M., Nierstrasz, O., Pandu Rangan, C., Steffen, B. et al. (eds.) *Advances in User Modeling*, pp. 105–116. Springer, Berlin Heidelberg (2012)
 32. Arnott, D., Pervan, G.: Design Science in Decision Support Systems Research. An Assessment using the Hevner, March, Park, and Ram Guidelines. *Journal of the Association for Information Systems* 13, 923–949 (2012)
 33. Kukreja, N., Boehm, B.: Integrating Collaborative Requirements Negotiation and Prioritization Processes: A Match Made in Heaven. In: *International Conference on Software and System Process*, pp. 141–145 (2013)
 34. Grünbacher, P., Köszegei, S., Biffli, S.: Stakeholder Value Proposition Elicitation and Reconciliation. In: Biffli, S., Aurum, A., Boehm, B., Erdogmus, H., Grünbacher, P. (eds.) *Value based software engineering*, pp. 133–154. Springer, Berlin Heidelberg (2006)
 35. Lenz, A., Schoop, M.: Assessment of Multi-Criteria Preference Measurement Methods for a Dynamic Environment. In: *Proceedings of the 52nd Hawaii International Conference on System Sciences*, pp. 1803-1812 (2019)
 36. Bergsmann, J.: *Requirements Engineering für die agile Softwareentwicklung. Methoden, Techniken und Strategien*. dpunkt, Heidelberg (2014)
 37. Engin, A., Vetschera, R.: Information representation in decision making: The impact of cognitive style and depletion effects. *Decision Support Systems* 103, 94–103 (2017)
 38. Gettinger, J., Koeszegei, S.T.: Far from Eye, Far from Heart. Analysis of Graphical Decision Aids in Electronic Negotiation Support. *Group Decis Negot* 23, 787–817 (2014)