New Developments for ROBERT – Assisting Novice Users Even Better in DIY Projects

Gregor Behnke,[1,3*] Pascal Bercher,[2*] Matthias Kraus,[1] Marvin Schiller,[4] Kristof Mickeleit,[1] Timo Häge,[1] Michael Dorna,[4] Michael Dambier,[4] Dietrich Manstetten,[4] Wolfgang Minker,[1] Birte Glimm,[1] and Susanne Biundo[1]

[1]Ulm University, Germany, firstname.lastname@uni-ulm.de
[2]The Australian National University, Australia, pascal.bercher@anu.edu.au
[3]University of Freiburg, Germany, behnkeg@cs.uni-freiburg.de
[4]Corporate Research Sector, Robert Bosch GmbH, Germany, firstname.lastname@de.bosch.com

Abstract

Do-It-Yourself (DIY) home improvement projects require a combination of specific knowledge and practical abilities. Novice users often lack both and thus tend to fail or be frightful of performing DIY projects – even though they would like to. By providing suitable and individualised assistance in the form of step-by-step instructions, the assistant ROBERT allows even novice users to successfully complete their DIY projects. Simultaneously, ROBERT allows its users to learn how to perform these steps themselves and thus enables them to become more independent in the future. In this paper, we report on the latest progress with ROBERT. Compared to earlier versions, ROBERT is now able to adaptively change its instructions based on the wishes and preferences of the user. Further, ROBERT is now able to use connected tools – i.e. tools that are able to sense and communicate their status – to check whether the user is performing the project's steps correctly and to provide further assistance in the case of failure. Lastly, we present the results of an empirical study conducted to show ROBERT's effectiveness.

Introduction

Performing handiwork tasks without the advice of an experienced person is practically very challenging. A prevalent type of such handiwork tasks are Do-It-Yourself (DIY) home improvement projects, where a person e.g. tries to construct a small object for his or her home (a bird house, a keyrack, ...) or to refurbish such objects. DIY novices are especially prone to fail in such projects – even though they would like to perform them in order to become more self-reliant or to save money. Individualised and adaptive assistance can play a central role in enabling novice users to successfully complete their projects by enabling users to learn about the tools, materials, and procedures involved. This assistance also must be individualised and adaptive as it has to take the specific tools and materials into account that the user actually possesses and has to make specific recommendations for them (e.g. how to position the gear switch of a specific drill). Systems that provide such a kind of assistance are called Companion Systems (Biundo et al. 2016).

*Most of the work was done while still at Ulm University.

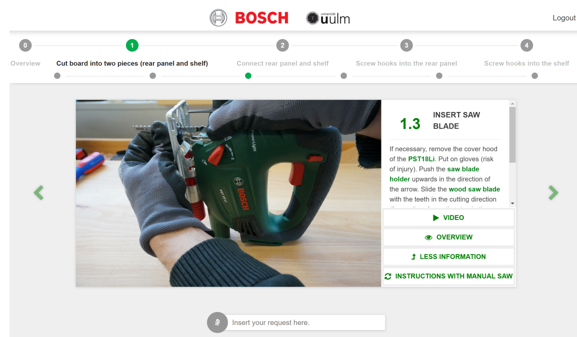

Figure 1: Robert explaining its user how to insert a saw blade into an electric saw. It also offers its user the option to switch to another tool for sawing.

The assistant ROBERT, which was developed in collaboration with Robert Bosch GmbH, one of the leading manufacturers of electric tools, provides such assistance to novice users (Behnke et al. 2019). For a given DIY project, ROBERT provides its user with a step-by-step instruction detailing on how to complete it successfully. These steps are generated by a planner and are thus adapted to the specific situation of ROBERT's user. ROBERT presents the individual instructions using text, images, voice and videos (see Fig. 1), which are selected automatically using description logic reasoning.

In this paper, we present the newest version of ROBERT, for which we added two important capabilities: First, ROBERT is now able to adapt its instructions, i.e. its plan, to the wishes stated by its users. Second, ROBERT can now use the data provided by a connected tool to sense which action its user is performing and can thus provide helpful assistance in case of problems.

ROBERT's Architecture

The work in this paper is based on an earlier version of the assistant ROBERT (Behnke et al. 2019; 2018). It comprises three components, which work together in order to provide suitable assistance to ROBERT's users: a planning-, an ontology-, and a dialogue-management-component.

ROBERT's planner is responsible for finding a course of action that completes the DIY project the user wants to undertake. The planner operates on a description of the available tools and materials and a general planning domain model, which contains actions describing the activities typically performed in DIY projects (sawing, drilling, sanding, fixating, marking, ...). This allows ROBERT's planner to adapt its instructions to the current situation, e.g. by recommending another means of making a large hole, if no Forstner bit is available. The planning model is formulated based on Hierarchical Task Network (HTN) planning (Erol, Hendler, and Nau 1996; Bercher, Alford, and Höller 2019), which is known to be more expressive than e.g. classical planning (Höller et al. 2014; Höller et al. 2016). Using the HTN, ROBERT can not only provide detailed instructions, but also more abstract instructions for users that are already familiar with some procedures in the DIY setting (e.g. predrilling). ROBERT uses a SAT-based planner to find optimal (shortest) plans (Behnke, Höller, and Biundo 2019b; 2019a; 2018b; 2018a), which are currently the best-performing HTN planners for this task. Search-based planners (Bercher et al. 2017; Höller et al. 2018) are currently not able to solve the complex planning problems faced by ROBERT.

The ontology component within ROBERT is responsible for managing the static knowledge in the DIY domain. This includes information like the colour and shape of objects (drills, bits, saw blades, . . .) as well as technical aspects of the tools involved (which battery type fits onto which device). Lastly, the ontology also stores information about recommended configurations for actions, e.g. the drilling speed for hard wood. The ontology manager packages and sends this information to both the planner and the dialogue manager. Further, the selection of media contents to be presented to the user is done by the ontology manager. The planner transmits information about the action to be displayed (name and parameters) and the ontology selects a *best-fitting* description and media content via *classification*. ROBERT's ontology manager is also responsible for generating textual explanations for factual knowledge requested by the user.

ROBERT's dialogue manager (Kraus et al. 2019) deals with all aspects of the communication between ROBERT and its user. After the user has selected his or her DIY project, the dialogue manager invokes the planner, which generates a plan providing appropriate instructions. The plan is transmitted back to the dialogue manager, which hands its steps to the ontology manager. It in turn augments the symbolic description of the actions with textual instructions and media contents. They are passed back to the dialogue manager, which presents the augmented instruction to its user. It allows the user to navigate through the plan using multimodal inputs (speech, touch). If the user poses questions to ROBERT, the dialogue management translates these requests into a symbolic representation and sends it to the responsible component (either the ontology manager, the planner, or the dialogue manager itself). That component determines an appropriate reaction to the user's request and transmits it to the dialogue manager, which in turns transmits it to the user.

All three components share the same model information. However, each component only stores the information for handling which it is best suited for. When required, information is transmitted from one component to another. To allow for this interoperability and to ensure coherent storage of models and information, we use a specific modelling paradigm, which e.g. allows for storing parts of the planning model in a structured way in the ontology (Schiller et al. 2017; Behnke et al. 2019).

Adapting Plans to the User's Wishes

One key challenge that has not been addressed by the former versions of ROBERT is to allow the user to control the results of the planning process and consequently the instructions he or she is presented with. This ability is especially important in order to make the process of generating the planning a collaborative one (Nothdurft et al. 2017; 2015). So far, ROBERT's planner simply generated *some optimal* plan for the DIY task at hand and presented it to its user. This ensured correct and situation-adaptive assistance. The user, however, had no influence over the presented plan. Especially in the DIY setting, this is problematic, as projects often allow for a multitude of options (e.g. regarding the means used to connect two wooden boards). As a notable example (which we also used in our user study), there are two main ways to split a wooden board into two: manual and electric sawing. Even if both means are technically possible, the planner will select the shorter plan, usually the one using the manual saw as it does not require to setup an electric tool. Users, however, tend to have a preference for one of the two options. This can be a personal predisposition – e.g. user may fear using an electric saw. A user may also want to perform the DIY project to learn about the tools involved in order to become more independent – this is the main motivation behind ROBERT's design. Forcing the user to use a specific means to achieve a goal (i.e. to use one saw or the other) hinders the latter goal or may even lead to the user aborting his interaction with the system (in the former case).

To give ROBERT the ability to adapt its plan to the wishes and preferences of its user, we had to decide how wishes and preferences are to be represented. A first option would be to categorise the requests into types and to implement dedicated algorithms to handle each of them. Theoretical investigations have shown, however, that handling such requests is as hard as planning itself (Behnke et al. 2016), which led us to consider a more general approach. ROBERT represents requests made by its user as formulae in Linear Temporal Logic (LTL) (Pnueli 1977). Notably, LTL Formulae also form the basis for the standard means to describe preferences in classical planning (Gerevini and Long 2005).

Given an LTL formula ϕ describing a user's wishes, we look for a plan that is both a solution to the planning problem, i.e. a plan that completes the DIY project, and that satisfies the formula ϕ. Since we use SAT-based planning, this amounts to adding further clauses to the formulae. Mattmüller and Rintanen (2007) have proposed such an encoding, which we use in a recently improved version (Behnke and Biundo 2018). It is unrealistic to assume that users are able to formulate their requests directly in terms of LTL formulae. Thus, ROBERT allows its users to input their requests in natural language and translates them

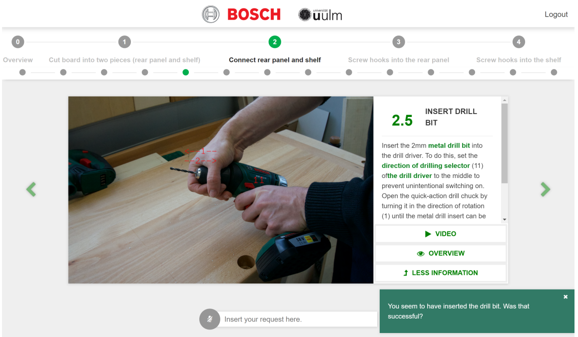

Figure 2: Instructions for inserting a metal drill bit including a proactive message for checking task performance.

automatically into LTL. For this, we use a pattern-based machine learning mechanism, like the one proposed by Nikora and Balcom (2009). For interactional purposes, ROBERT always finds some optimal plan first and presents it (as instructions) to the user. The user may – at any time – request a modification, which is translated into LTL and triggers a new planning process. LTL formulae from previous requests are kept and the new one is added on top. If this results in an unsolvable planning problem (e.g. by requiring to use a manual saw and not to saw at all), LTL constraints are dropped, starting with the oldest one, until a plan is found again.

Connected Tools

So far, ROBERT was an application that resided on a device (a tablet) with which the user interacted. ROBERT had no other means of interacting with its environment, hindering its ability to suitably react to the user handling tools and to possibly even intervene. We present here the – to the best of our knowledge – first industry-made tool, a connected drill, that is able to interact with an assistant. This connection allows ROBERT to provide the user with proactive assistance. To proactively provide assistance, we make use of sensor data for tracking the user's current activity, e.g. drilling or screwing. Therefore, an inertial measurement unit (IMU), capable of measuring gyroscopic, accelerometric and compass data, is integrated in a standard cordless screwdriver together with a development board with Wi-Fi connectivity for transferring measurements. For classification of the activities based on the transferred sensor data, ROBERT uses a neural network, which has been trained with data from 12 participants. It allows for differentiating six user activities: *off*, *screwing*, *drilling*, *drill change*, *battery change*, and *other*. Additional information is available in the form of a probability distribution e.g. of the current activity, the activity's operation time and frequency of its occurrence.

In this work, proactive system behaviour addresses the active initiation of an interaction with the user to check whether he or she is performing the project's steps correctly and to provide help in the case of failure. Therefore, we implemented a rule-based decision model for when and how to intervene. At each project step, where the connected tool needs to be applied, ROBERT is able to actively start an interaction with the user. Depending on the context it is differentiated between three different interaction scenarios.

Finally, ROBERT checks whether the user performs the instructed task correctly (see Fig. 2). An interaction is initiated whenever one of the activities *battery change*, *drilling*, *drill change*, or *screwing* is recognised. The user is asked whether the current step is going well. If the user answers with *no*, further help is offered. If help is requested, the system starts an explanatory video of the task. Otherwise, the system apologizes for the disturbance. Here we give an example dialogue of this interaction scenario between the system (**S**) und its user (**U**):

S: I noticed that you were drilling. Did that work?
U: No, it didn't. **S:** Ok, do you need additional help?
U: Yes, please. **S:** A video of this project step could help. I'm going to play it for you. *(Then a video is played.)*

For making the user aware of the connected tool, a first message is triggered after the user picked up the machine: *"You seem to be working with the connected tool for the first time. Don't worry, I'll guide you through the steps!"*. This message can be triggered only once. In order to react to possible user insecurity about the current project step, a help request is sent to the user after three minutes of inactivity during a step, where an action with the connected tool has to be performed: *"I haven't seen any tool activity by you in three minutes. Do you need help?"*. If the user affirms this question, he or she is invited to watch a video of the current step.

Evaluation

A first empirical study with the previous version of ROBERT was conducted in 2018 and yielded promising results (Schiller et al. 2018; Behnke et al. 2019). The goal of the current study was to investigate the effect of plan adaptation and the use of connectivity for pro-active user support. We used an A/B-test design (n=32), where ROBERT was provided in a pro-active version (using the connected tool) and a re-active version (n=16 in both conditions). The recruited participants were DIY novices with a balanced split of gender and age groups. As in the previous study, participants were first introduced to the study and ROBERT, then they filled in a pre-study questionnaire, which assessed demographic data and previous experience with DIY and tools. They were then asked to construct a keyrack from a wooden plank together with ROBERT. Finally, the participants filled in a post-study questionnaire, which assessed the overall verdict and various specific dimensions, e.g. relating to the option to change the instructions.

For the step of sawing a plank, ROBERT offered all participants (in both groups) the option to switch between an electric jigsaw and a crosscut handsaw, adjusting the necessary sub-steps (inserting a sawblade and a battery into the saw) accordingly on the fly using LTL planning. This option was made explicit to ROBERT's users via a button in the graphical user interface (cf. Fig. 1). We refrained from allowing the participants *any* further change to the plan, in order to keep the study controllable. Our study aims at investigating how participants used this adaptivity (and whether they appreciate this feature), which can be observed only when

Table 1: Participants' scores on questions (interpolated median (IM), mean & standard deviation (SD) on a Likert scale 1: do not agree at all – 5: fully agree, for the overall verdict 1: inadequate – 5: very good), translated from German

General Questions			
	overall	proactive	non proactive
Verdict (IM; mean (SD))	3.23; 3.17 (0.9)	3.38; 3.13 (0.96)	3.14; 3.2 (0.83)

The app allowed for switching the tool. In how far do you agree with the following statements in this regard?			
	overall	no change	tool change
The capability of adjusting the instructions to my individual preferences (e.g. preferred tool) is important to me.	4.53; 4.24 (0.93)	3.5; 3.7 (1.0)	4.71; 4.53 (0.75)
The instructions would be (even) better, if the opportunity was offered more often to adjust them to my preferences.	3.42; 3.28 (1.28)	4.0; 3.5 (1.57)	3.2; 3.16 (1.09)
Due to switching the tool, I received the instructions that I like.			4.4; 4.00 (1.17)
The switch of tools was carried out in correspondence with my expectations.			4.29; 4.05 (1.10)

the participants have a reason to change (rather than simply follow) the instructions. In the pre-study questionnaire (presented on a separate computer) we therefore included a forced-choice question about participants' preference for either the jigsaw or the handsaw. Unknown to the participants, ROBERT was remote-controlled to "incidentally" present an initial plan with the non-preferred option to all participants alike (except one, due to a mistake), allowing us to observe the participants' propensity to accept the offered change of plan. Apart from plan changes, the participants were asked to avoid deviations from the instructions. In the following, we exclude two participants (one each from groups A and B) from the analysis who did not initiate a plan change but proceeded to use their preferred tool while outright ignoring ROBERT's instruction (leaving n_A=n_B=15). Statistical tests were performed with R (R Core Team 2018).

As noted above, the A/B-test served to evaluate ROBERT's new pro-active abilities. When comparing the pro-active version with the re-active version, we did not find any significant difference (with alpha=.05 for all statistical tests) in participants' replies on the post-questionnaire dimensions (navigation & design, trust, acceptance, usefulness & understandability, reliability, competence, predictability & transparency, faith, personal attachment, and interaction). The overall verdict about ROBERT (on a Likert scale of 1: inadequate – 5: very good) was also not improved by pro-activity (see Tab. 1). Participants who switched the tool (n_1=19) rated ROBERT slightly more positively at 3.32 (SD=0.98) on average (interpolated median IM=3.56) vs. those who did not (n_2=10) at 2.8 (SD=0.6, IM=2.83) (totals exclude the session where the preferred tool was accidentally set), but the difference fails to reach statistical significance (Mann-Whitney U-Test, U=130, $p = .095$ two-tailed). Compared to the previous version of ROBERT, the overall evaluation fell from a mean value of 3.89 (Behnke et al. 2018). One reason for the decrease might be that ROBERT initially provided the user with the instructions he did not prefer – even though this selection was unknown to the participants. However, we observed gender differences in the assessment of ROBERT. We noted that in spite of being recruited for the study according to the same criteria, the subgroups of female and male participants were not matched in their DIY experience. Overall the female group turned out to be less experienced in comparison to males. This was also reflected in their answers for instance to our question whether ROBERT did not provide enough information about tools and materials, where females tended to agree and males tended to disagree, yielding a significant difference (Mann-Whitney U-Test, U=195.5, n_f=16, n_m=14, $p < .001$ two-tailed). Thus we observed further potential to adapt ROBERT more specifically to different target groups within self-reported DIY novices.

During the experiments, 19 participants made use of a change of plan vs. 10 who did not (not counting the two excluded participants and the one session where the preferred tool was accidentally shown from the start). We conjecture that it is important to provide users the possibility to change plans according to their preferences. Out of the 19 participants, most (16) indicated a preference for the electric jigsaw, were then shown a plan with the handsaw, and changed it. The post-study questionnaire asked participants whether they consider the possibility to adapt the instructions according to their preferences important (on a five-point Likert scale 1 – 5), yielding a high average agreement of 4.24 (IM=4.53). As expected, those who switched the tool agreed significantly more (average=4.53, IM=4.71) than those who did not (Mann-Whitney U-Test, U=138.5, n_1=19, n_2=10, $p = .031$ two-tailed). Table 1 shows participants' answers to the planning-related questions. Note that the third and fourth question apply only to participants who changed the tool. We gave the participants the option to provide a textual evaluation of the ability to adapt the instructions. In total 18 participants answered. Six participants voiced a positive opinion ("useful", "helpful"), two a negative opinion ("not useful for me", "not necessary"), and three a neutral opinion ("ok"). The seven remaining answers did not provide an explicit preference, but four of those participants provided a reason why or why not they changed the plan.

Conclusion

We presented recent progress in the development of the interactive assistant ROBERT, which supports users in performing Do-It-Yourself projects. We have shown how users can control the instructions they are given using LTL planning, which allows for adapting to the users' preferences. In our empirical study, the participants perceived this feature as positive. Lastly, ROBERT is now able to use sensor-data from a connected tool to provide situation adaptive feedback to users on their performance in individual steps.

Acknowledgments

This work is done within the technology transfer project "Do it yourself, but not alone: Companion-Technology for DIY support" of the Transregional Collaborative Research Centre SFB/TRR 62 "Companion-Technology for Cognitive Technical Systems" funded by the German Research Foundation (DFG). We thank MTO Psychologische Forschung und Beratung GmbH for facilitating the reported experiments. We thank three reviewers for helpful comments. We thank Florian Schiller for additional advice.

References

Behnke, G., and Biundo, S. 2018. X and more parallelism: Integrating LTL-next into SAT-based planning with trajectory constraints while allowing for even more parallelism. *Inteligencia Artificial* 21(62):75–90.

Behnke, G.; Höller, D.; Bercher, P.; and Biundo, S. 2016. Change the plan – How hard can that be? In *Proc. of the 26th Int. Conf. on Autom. Plan. and Sched. (ICAPS 2016).*

Behnke, G.; Schiller, M.; Kraus, M.; Bercher, P.; Schmautz, M.; Dorna, M.; Minker, W.; Glimm, B.; and Biundo, S. 2018. Instructing novice users on how to use tools in DIY projects. In *Proc. of the 27th Int. Joint Conf. on AI and the 23rd Europ. Conf. on AI (IJCAI-ECAI 2018).*

Behnke, G.; Schiller, M.; Kraus, M.; Bercher, P.; Schmautz, M.; Dorna, M.; Dambier, M.; Minker, W.; Glimm, B.; and Biundo, S. 2019. Alice in DIY-wonderland or: Instructing novice users on how to use tools in DIY projects. *AI Communications* 32(1):31–57.

Behnke, G.; Höller, D.; and Biundo, S. 2018a. totSAT – Totally-ordered hierarchical planning through SAT. In *Proc. of the 32nd AAAI Conf. on AI (AAAI 2018).*

Behnke, G.; Höller, D.; and Biundo, S. 2018b. Tracking branches in trees – A propositional encoding for solving partially-ordered HTN planning problems. In *Proc. of the 30th Int. Conf. on Tools with Art. Int. (ICTAI 2018).*

Behnke, G.; Höller, D.; and Biundo, S. 2019a. Bringing order to chaos – A compact representation of partial order in SAT-based HTN planning. In *Proc. of the 33rd AAAI Conf. on AI (AAAI 2019).*

Behnke, G.; Höller, D.; and Biundo, S. 2019b. Finding optimal solutions in HTN planning – A SAT-based approach. In *Proc. of the 28th Int. Joint Conf. on AI (IJCAI 2019).*

Bercher, P.; Alford, R.; and Höller, D. 2019. A survey on hierarchical planning – One abstract idea, many concrete realizations. In *Proc. of the 28th Int. Joint Conf. on AI (IJCAI 2019).*

Bercher, P.; Behnke, G.; Höller, D.; and Biundo, S. 2017. An admissible HTN planning heuristic. In *Proc. of the 26th Int. Joint Conf. on AI (IJCAI 2017).*

Biundo, S.; Höller, D.; Schattenberg, B.; and Bercher, P. 2016. Companion-technology: An overview. *Künstliche Intelligenz* 30(1):11–20. Special Issue on Companion Technologies.

Erol, K.; Hendler, J.; and Nau, D. 1996. Complexity results for HTN planning. *Annals of Mathematics and AI* 18(1):69–93.

Gerevini, A., and Long, D. 2005. Plan constraints and preferences in PDDL3. Technical report, Department of Electronics for Automation, University of Brescia.

Höller, D.; Behnke, G.; Bercher, P.; and Biundo, S. 2014. Language classification of hierarchical planning problems. In *Proc. of the 21st Europ. Conf. on AI (ECAI 2014).*

Höller, D.; Behnke, G.; Bercher, P.; and Biundo, S. 2016. Assessing the expressivity of planning formalisms through the comparison to formal languages. In *Proc. of the 26th Int. Conf. on Autom. Plan. and Sched., (ICAPS 2016).*

Höller, D.; Bercher, P.; Behnke, G.; and Biundo, B. 2018. A generic method to guide HTN progression search with classical heuristics. In *Proc. of the 28th Int. Conf. on Autom. Plan. and Sched. (ICAPS 2018).*

Kraus, M.; Schiller, M.; Behnke, G.; Bercher, P.; Biundo, S.; Glimm, B.; and Minker, W. 2019. A multimodal dialogue framework for cloud-based companion systems. In *Proc. of the 9th Int. Work. on Spoken Dialogue Systems Technology (IWSDS 2019).*

Mattmüller, R., and Rintanen, J. 2007. Planning for temporally extended goals as propositional satisfiability. In *Proc. of the 20th Int. Joint Conf. on AI (IJCAI 2007).*

Nikora, A., and Balcom, G. 2009. Automated identification of LTL patterns in natural language requirements. In *Proc. of the 20th IEEE Int. Conf. on Software Reliability Engineering (ISSRE 2009).*

Nothdurft, F.; Behnke, G.; Bercher, P.; Biundo, S.; and Minker, W. 2015. The interplay of user-centered dialog systems and AI planning. In *Proc. of the 16th Ann. Meet. of the SIG on Discourse and Dialogue (SIGDIAL 2015).*

Nothdurft, F.; Bercher, P.; Behnke, G.; and Minker, W. 2017. User involvement in collaborative decision-making dialog systems. In Jokinen, K., and Wilcock, G., eds., *Dialogues with Social Robots: Enablements, Analyses, and Evaluation*. This book chapter was accepted at the 7th International Workshop On Spoken Dialogue Systems (IWSDS 2016).

Pnueli, A. 1977. The temporal logic of programs. In *Proc. of the 18th Ann. Symp. on Foundations of Comp. Sci. (SFCS 1977).*

R Core Team. 2018. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria.

Schiller, M.; Behnke, G.; Schmautz, M.; Bercher, P.; Kraus, M.; Dorna, M.; Minker, W.; Glimm, B.; and Biundo, S. 2017. A paradigm for coupling procedural and conceptual knowledge in companion systems. In *Proc. of the 2nd Int. Conf. on Companion Tech. (ICCT 2017).*

Schiller, M.; Behnke, G.; Bercher, P.; Kraus, M.; Dorna, M.; Richter, F.; Biundo, S.; Glimm, B.; and Minker, W. 2018. Evaluating knowledge-based assistance for DIY. In Dachselt, R., and Weber, G., eds., *Proc. of MCI Works. "Digital Companion".*