TECHNICAL REPORT
Retrieving Adaptable Cases in Process-Oriented Case-Based Reasoning

Ralph Bergmann and **Gilbert Müller** and **Christian Zeyen** and **Jens Manderscheid**

Department of Business Information Systems II
University of Trier
54286 Trier, Germany
http://www.wi2.uni-trier.de
{bergmann}{muellerg}{s4chzeye}{s4jemand}@uni-trier.de

Abstract

This paper presents a novel approach to retrieval in process-oriented case-based reasoning (POCBR) which considers the adaptability of workflows cases during the retrieval phase. A novel concept of adaptability in POCBR is proposed, which assesses the potential similarity increase of a case which can be gained by adaptation. The adaptability of a case is learned from the case base in an off-line pre-processing phase prior to the retrieval. The proposed approach is generic as it can be used in combination with different adaptation methods. An empirical evaluation in the domain of cooking workflows demonstrates the benefit of the approach.

Introduction

Case-based reasoning (CBR) is a well-known approach to problem solving and learning on the basis of experience made in the past (Aamodt and Plaza 1994; Lopez de Mantaras et al. 2005). Traditionally, experience is captured in the form of cases, which are problem-solution pairs stored in the case base. The basic assumption of CBR is that similar problems have similar solutions. Thus, a new problem is solved by retrieving cases addressing similar problems than the one stated in the query. Then, the solution of the retrieved case is adapted to compensate the differences in the problem description between the retrieved case and the query. Various different methods have been investigated for adaptation (Lopez de Mantaras et al. 2005). Adaptation can be performed by substituting isolated parts of the solution, by more comprehensive transformations including changing the structure of the solution, by generalization and subsequent specialization, or even by combining solution fragments coming from several similar cases. A common characteristic of all adaptation approaches is the fact that domain specific adaptation knowledge is required, which often leads to a knowledge acquisition bottleneck, recently being addressed by applying machine learning methods (Hanney and Keane 1997; Craw, Wiratunga, and Rowe 2006). As a consequence, the problem solving capability strongly depends on the applied adaptation method and the available adaptation knowledge. In particular, the coverage (Smyth and Keane 1995) of each single case in the case base, i.e., the set of problems for which it can be successfully reused by adaptation is determined by the adaptation capabilities. For solving a problem in a case-based manner this has immediate consequence for retrieval as well. The ultimate goal of retrieval should be to select cases which can be best adapted to become a solution to the current problem. The basic CBR principle claims that this should be the most similar case. However, various research in the past has demonstrated that this is not necessarily the case (Smyth and Keane 1994; Leake, Kinley, and Wilson 1997). Instead of retrieving the most similar case, the most useful case (Bergmann et al. 2001) should be retrieved, while utility is defined with respect to adaptability for the current query. In this view, similarity turns into a means for approximating the utility of a case, i.e., to assess the degree by which a case can be adapted before actually performing

This paper is accepted for publication at the 29th International FLAIRS Conference, Key Largo (Florida), USA. AAAI, 2015

the (often quite computationally expensive) adaptation. However, the majority of standard similarity measures does not consider this relationship sufficiently. In particular, the definition of similarity must be aligned with the adaptation method and the currently available adaptation knowledge. The just explained dependency between adaptation and retrieval has been investigated to some extent in previous research. Proposed solutions include adaptation guided retrieval proposed by Smyth and Keane (1994) as part of their CBR system Déjà Vu. Another approach was proposed by Leake et al. (1997) in which adaptability is estimated from experience, leading to an improved similarity assessment based on re-application cost and relevance of cases. A different approach focusses on learning of similarity measures (Stahl 2004) based on feedback from adaptation experience. Thereby, the similarity measure is trained to consider the adaptability, which can improve the retrieval results.

Previous work on retrieval of adaptable cases is mainly focussed on attribute-value or object-oriented representations. In this paper, we investigate this issue within process-oriented CBR (POCBR) (Minor, Montani, and Recio-Garca 2014), which deals with CBR applications for process-oriented information systems. POCBR aims at assisting domain experts in their work with workflows, in particular by supporting workflow reuse. Two important problems of workflow reuse are the retrieval of similar workflows from potentially large repositories (Bergmann and Gil 2014) as well as the adaptation of workflows (Müller and Bergmann 2014). Adaptation is particularly important in POCBR as the number of available cases is usually not very large. Recently, several adaptation methods for POCBR have been proposed (Müller and Bergmann 2014; 2015a; 2015b). Thus, the ability to retrieve adaptable cases is of increasing importance.

This paper presents the first approach for retrieving adaptable cases which is applicable for POCBR. Following the basic idea of Leake et al. (1997) we aim at estimating the adaptability of a workflow case based on the experience within the case base. In an offline phase, the case base is automatically analyzed by performing and monitoring adaptation executions using the available adaptation knowledge. Based on these results, the adaptability of each case is assessed and stored together with the case. Further, retrieval is extended to not just use available similarity measures but in addition also the adaptability rating of the case under investigation. Two different methods for adaptability assessment are proposed. The first method determines a global adaptability score applied for all queries while the second method determines adaptability in a query-specific manner.

In the next section, we present the necessary foundations of process-oriented CBR as well as an example from the domain cooking recipes demonstrating the need for adaptation-guided retrieval. Then, the developed approaches are described in detail. An evaluation is performed using a case base of pasta recipes and a compositional adaptation approach for workflows (Müller and Bergmann 2014). A discussion of related work and future research closes this paper.

Foundations

We now briefly introduce relevant previous work in the field of POCBR and illustrate it in the domain of cooking recipes, which are represented as workflows.

Representation of Semantic Workflow Cases

Broadly speaking, a workflow consists of a set of *activities* (also called *tasks*) combined with *control-flow structures* like sequences, parallel (AND split/join) or alternative (XOR split/join) branches, and loops. Tasks and control-flow structures form the *control-flow*. In addition, tasks exchange certain *products*, which can be of physical matter (such as ingredients for cooking tasks) or data. Tasks, products, and relationships between the two of them form the *data flow*. Today, graph representations for workflows are widely used. We use a workflow representation based on semantically labeled graphs (see Fig. 1) as introduced by Bergmann and Gil (2014).

Definition 1 *A workflow is a directed graph $W = (N, E, S, T)$ where N is a set of nodes and $E \subseteq N \times N$ is a set of edges. The function T assigns each node and edge a type. Further, nodes have a semantic description from a semantic meta data language Σ, which is assigned by the function $S : N \to \Sigma$.*

For this work, Σ is defined by domain specific light-weight ontologies, restricted to a taxonomical representation of terms. In particular, we use one ontology for tasks and one ontology for data items.

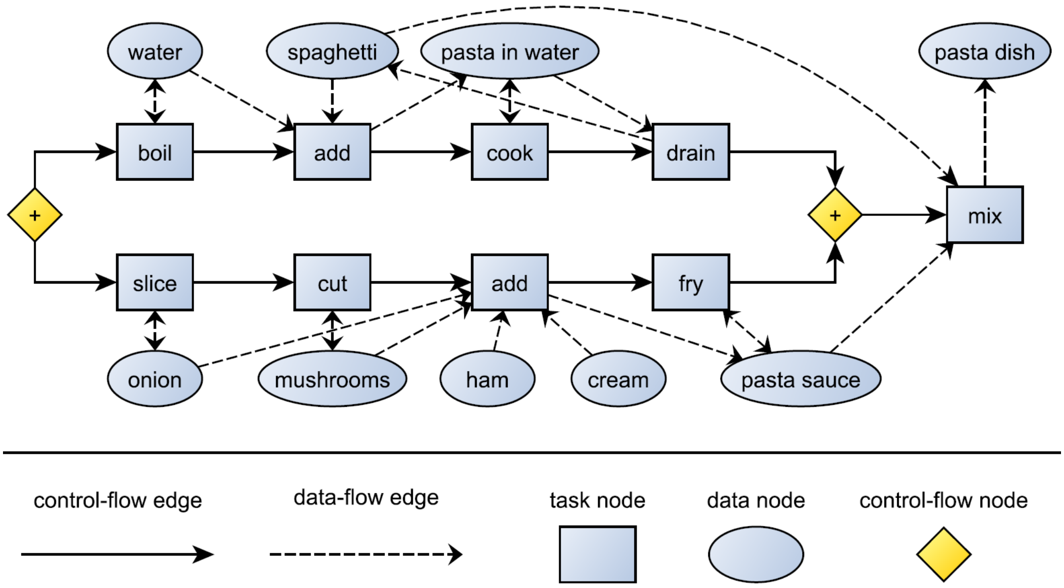

Figure 1: Simple workflow case representing a pasta recipe

In the cooking domain, the task ontology organizes the various cooking steps in a taxonomical order and the data ontology represents the ingredients.

Figure 1 shows a workflow graph from the cooking domain with different types of nodes and edges. The task nodes and data nodes represent tasks and data items, respectively. The data-flow edge is used to describe the linking of the data items consumed and produced by the tasks. The control-flow edge is used to represent the control flow of the workflow, i.e., it links tasks with successor tasks or control-flow elements.

Queries in Process-Oriented Case-Based Reasoning

Queries in POCBR describe user requirements on the desired workflow to be computed in a case-based manner, i.e., by adapting the retrieved most similar workflow case. For this purpose, we use a restricted form of POQL (Query Language for Process-Oriented Case-Based Reasoning) (Müller and Bergmann 2015c) which enables to represent desired and undesired nodes of a workflow, for example ingredients or preparation steps. Let $q_d = \{x_1, \ldots, x_n\}$ be a set of desired nodes and $q_u = \{y_1, \ldots, y_n\}$ be a set of undesired nodes. A query q is then defined as $(x_1 \wedge \ldots \wedge x_2) \wedge \neg y_1 \wedge \ldots \wedge \neg y_n$. POQL also enables to capture generalized terms, i.e., if a vegetarian dish is desired, this can be defined by $\neg meat$. POQL also allows to include control and data-flow links between the nodes, i.e. full sub-workflows can be used to specify desired and undesired workflow structures, however in this paper we don't support this feature of POQL.

The query q is used to guide retrieval, i.e., to search for a workflow which at best does not contain any undesired element and contains all desired elements. Based on the query q the unmatched elements can be identified. Thereby it is determined which elements should be deleted from or added to the retrieved workflow during the adaptation stage.

Semantic Similarity

In our previous work, we developed a semantic workflow similarity framework Bergmann & Gil (2014) which allows to assess the similarity between two workflows. The similarity model is based on a local similarity measure sim_Σ for the terms from the semantic meta data language (ontology) Σ assigned to the nodes of the workflow. We apply the taxonomy similarity approach by Bergmann (1998) to model the similarity between two ontology nodes based on their closeness. The similarity $sim_N : N^2 \rightarrow [0,1]$ of two nodes is then defined based on sim_Σ, i.e.,

$sim_N(n_1, n_2) = sim_\Sigma(S(n_1), S(n_2))$. The similarity between a query workflow and a case workflow is defined by means of an admissible mapping $m : N_q \cup E_q \rightarrow N_c \cup E_c$, which is a type-preserving, partial, injective mapping function of the nodes and edges of query workflow to those of case workflow. This means that nodes are only mapped to nodes of the same type and edges are only mapped if their corresponding nodes are mapped as well. Partial means that not all nodes of the case workflow must occur in the image of the mapping. For each query node x mapped by m, the similarity to the respective case node is computed by $sim_N(x, m(x))$. The overall workflow similarity with respect to a mapping m, named $sim_m(QW, CW)$ is then computed by an aggregation function (we use the average) combining the previously computed similarity values. Finally, the overall workflow similarity is determined by the best possible mapping m, i.e., $sim_{WF}(QW, CW) = \max\{sim_m(QW, CW) \mid \text{admissible map}\, m\}$. Thus, similarity assessment is defined as an optimization problem aiming at finding the best possible mapping, reflecting the best possible way to reuse the case workflow. Please note that the similarity measure is not symmetric as it is based on a mapping of the query workflow to the case workflow.

For case retrieval using POQL queries, this similarity measure is modified to consider also undesired workflow nodes. In this paper we use a simplified version of it, as queries only contain lists of desired and undesired nodes rather than fully linked workflow graphs. The best mapping m is computed for the set of desired nodes q_d (considering the nodes as isolated nodes in a query graph) as well as for each undesired node, mapping it to the most similar node in the case workflow.

The similarity $sim(q, CW)$ between a POQL query q and a case workflow CW is defined as the similarity between the desired nodes and the workflow CW and the number of undesired nodes not contained in CW according to the semantic similarity measure in relation to the size of the query:

$$sim(q, CW) = \frac{\sum_{x \in q_d} sim_N(x, m(x)) + |\{y \in q_u | sim_N(y, m(y)) \neq 1\}|}{|q_d| + |q_u|} \quad (1)$$

For POCBR sim can be used as similarity measure for the retrieval of a workflow case for a given query, which is then used for adaptation. The aim of adaption can be considered to make the retrieved workflow more similar to the query, e.g. by substituting not perfectly matching nodes with ones that are more similar to the query or by removing certain parts of the workflow which contain undesired nodes.

Demand for Adaptation-Guided Retrieval

The proposed similarity measure sim (1) is just based on the domain ontologies and does not consider the available adaptation knowledge. This could lead to the well known problem, that a non-optimal case is selected for adaptation. For example, consider a case base consisting of pizza and pasta recipe workflows (see Fig. 1 as an example for a spaghetti recipe) as well as a query containing the ingredients tomato, tuna, and the preparation step cook. If the case base contains a recipe for a pizza with tuna and tomato, this recipe would be more similar than the spaghetti recipe in Fig. 1. However, the pizza recipe would not contain the preparation step cook but the step bake instead. Adaptation will quite likely fail if there is no adaptation knowledge changing recipes from baking to cooking, which is a more severe modification. On the other hand, the pasta recipe (which already includes the cook step) can be more easily changed by replacing the ingredients as desired.

Considering Adaptability During Retrieval

We now describe two variants of a generic approach for adaptation guided retrieval in POCBR. We assume that an adaptation component and the respective adaptation knowledge is available. Our approach treats the available adaptation method as black box. It performs introspective adaptation experiments using the adaptation component to learn an adaptability score for each case. This black-box use of adaptation has the potential advantage that this approach can be used with every adaptation method because it does not rely on a specific representation of cases and adaptation knowledge. Alternatively, one could also treat the adaptation in a white box fashion by directly analyzing the adaptation knowledge applicable to a case. While such an approach could in principle allow to assess the adaptability in a more fine-grained manner, it would be specific to the representation of the adaptation knowledge and thus would have to be adapted whenever a new adaptation method is used.

During the course of developing the method described here, we also investigated this idea to some extend, but did not yet succeed to produce satisfactory adaptability scores this way. Thus, this idea is not followed any further in this paper.

For adaptation guided retrieval the workflow case with the highest similarity is no longer chosen. Instead the workflow with the highest similarity after considering a potentially additional similarity gain by adaptation is selected.

For this purpose, we propose a new *adaptability score* $\mathrm{A}(q, C) \in [0, 1]$. In the retrieval phase for a given query the adaptability score is considered in addition to the similarity sim between the q and the cases C of the case base. This is reflected in the following definition of the utility of a case C for a query q:

$$utility(q, C) = sim(q, C) + (1 - sim(q, C)) \cdot \mathrm{A}(q, C) \tag{2}$$

In equation 2 the second term defines the additional similarity gain that can be potentially achieved by adaptation. Thus if $sim(q, C) = 1$ holds, the similarity cannot be increased anymore by adaptation. If $sim(q, C) < 1$ the similarity difference to 1 can be potentially compensated by adaptation. The adaptability $\mathrm{A}(q, C)$ specifies how well a case can be adapted. Like similarity, the adaptability value must be normalized to the interval $[0, 1]$. An adaptability of 1 means that the case can be fully adapted to the query such that it matches perfectly after adaptation. An adaptability of 0 means that it cannot be adapted at all. During retrieval instead of selecting the most similar case with respect to sim we now retrieve the most useful case with respect to $utility$.

While the above definitions are not restricted to POCBR, we now focus on cases represented as workflows. In the following we describe two approaches for computing adaptability scores: a global score which is independent from the query and a query-specific score which makes use of the parameter q in $\mathrm{A}(q, C)$.

Adaptability Estimation From the Case Base

In a nutshell, adaptability estimation from the case base means that for each workflow in the case base several different adaptations are determined (see Fig. 2). For each adaptation performed, the resulting similarity gain is measured and used to estimate the adaptability of the workflow case. In this process, the case base is also used to determine adaptation directions, assuming the case base is representative also for future queries.

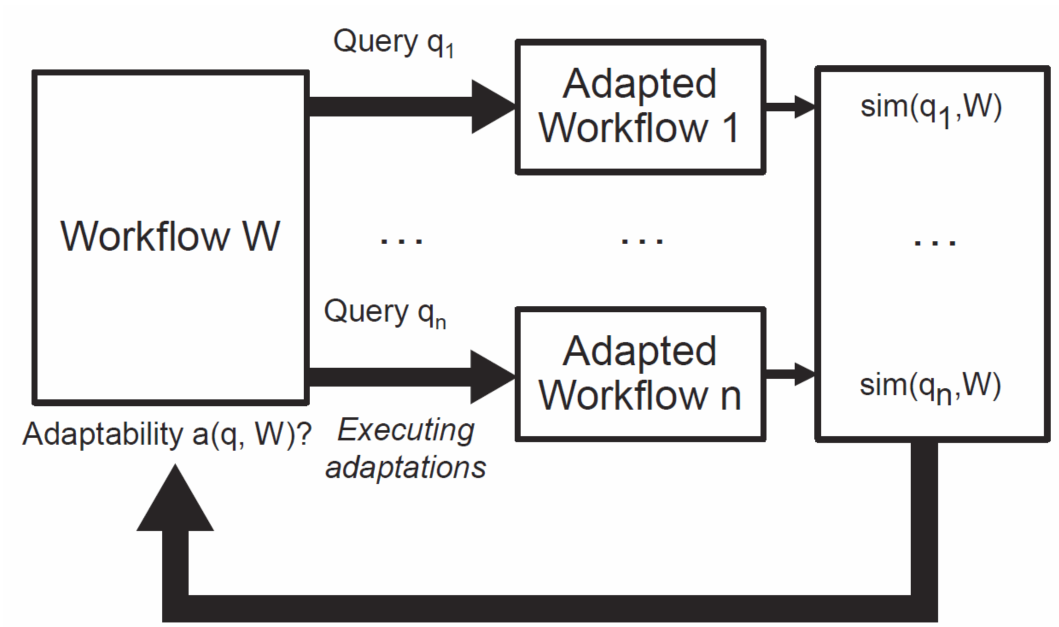

Figure 2: Adaptability estimation

More precisely, we attempt to adapt each workflow W_i in the case base CB having each other workflow $W_j \in \mathrm{CB}, i \neq j$ as the adaptation goal. Thus W_i is adapted with the goal that the nodes in W_i not occurring in W_j are removed and that the nodes in W_j not occurring already in W_i are added. This difference between W_i and W_j is reflected in a POQL query named q_{ij} that containts the respective desired and undesired nodes. For example, a query containing the undesired node "tomatoes", means that the workflow W_i uses tomatoes while the workflow W_j does not. The POQL query q_{ij} is then used as input to the adaptation method which uses it as a goal during the selection of the performed adaptation steps that are applied to W_i. The resulting workflow after adaptation is denoted as $\widehat{W_{ij}}$.

After each adaptation we compute the similarity gain, i.e. $\mathrm{sim}(q_{ij}, \widehat{W_{ij}}) - \mathrm{sim}(q_{ij}, W_i)$. If adaptation does not worsen the fit of the case with the query (which is not the case in the adaptation methods we apply), the similarity gain is never negative. Based on the similarity gain we determine the adaptability score according to equation (3). Thus, adaptability becomes a relative similarity gain aiming at estimating the degree in which adaptation can compensate the difference to a perfectly matching solution. We also call $\mathrm{a}(w_i, q_{ij})$ the *local adaptability* value as it is dependent from the query.

$$\mathrm{a}(W_i, q_{ij}) = \frac{\mathrm{sim}(q_{ij}, \widehat{W_{ij}}) - \mathrm{sim}(q_{ij}, W_i)}{1 - \mathrm{sim}(q_{ij}, W_i)} \tag{3}$$

Global Adaptability (GA)

Global adaptability approach (abbreviated as GA) is based on the assumption that adaptability is a property of a workflow case caused by the impact of the adaptation knowledge on the case which is not (strongly) dependent from the query. Thus we assume that adaptability is independent from the query and estimate the global adaptability $\mathrm{A}(q, W) \rightarrow [0, 1]$ of a case W based the observed local adaptability values by computing an average value (see equation 4). The global adaptability of a case is stored along with the case in the case base and used during retrieval to determine the individual utility of a case according to equation (2).

$$\mathrm{A}(\cdot, W_i) = \mathrm{A}(W_i) = \frac{\sum_{j=1}^{|CB|} \mathrm{a}(W_i, q_{ij})}{|CB|} \tag{4}$$

Query Specific Adaptability (QSA)

The query specific adaptability approach (abbreviated as QSA) does not make the assumption that adaptability is only dependent from the case. Instead it assesses the adaptability of a case during retrieval in a query-specific manner. We expect that thereby the adaptability can be assessed more precisely compared to the global approach. In this approach we store for each workflow case W_i in the case base an additional table (see Table) which represents all the computed local adaptability values $\mathrm{a}(W_i, q_{ij})$ together with the respective goal workflow W_j, i.e., the workflow which provided the adaptation goal during the pre-processing phase.

goal workflow W_j	local adaptability
W_1	$\mathrm{a}(W_i, q_{i1})$
...	
W_n	$\mathrm{a}(W_i, q_{in})$

Table 1: Local adaptability value table for case W_i

The idea of query specific adaptability is to estimate the adaptability value for each workflow in the case base, based on the most similar pre-computed adaptation scenario w.r.t. the current query. Thus, we select from this table the adaptability value for which the corresponding goal workflow is most similar to the query.

To summarize, retrieval using local adaptability works as follows: for each case in the case base the utility according to formula (2) is computed, consisting of the original similarity value and the local adaptability value. To compute the local adaptability, for each case in the case base an additional similarity-based retrieval is performed to find the most similar row in the case-specific table from which the adaptability value can be extracted. As similarity measure for this purpose we again use the POQL-query similarity as described in formula (1).

Obviously, this approach significantly increases the retrieval time as the retrieval complexity increases from $O(n)$ to $O(n^2)$ with n being the size of the case base. This limits the applicability of this approach when the case base is getting larger. In our experiments reported in the next section we use this approach also as a target for comparison to assess the benefit of the global approach. The limitations of this approach are discussed in the final section of this paper.

Evaluation

The GA and QSA approach to adaptation-guided retrieval of workflows have been fully implemented as part of the CAKE framework[1], which already contains the required retrieval and adaptation capabilities (Bergmann et al. 2014). We experimentally evaluated GA and QSA compared to the standard similarity-based retrieval (referred to as SIM) using the similarity measure in formula (1) to investigate two hypotheses.

Hypothesis 1 *Adaptation guided retrieval with GA and QSA is able to select cases which can be adapted more widely compared to the cases selected by SIM. Further, QSA outperforms GA in this respect.*

Hypothesis 2 *Case-based reasoning using adaptation guided retrieval with GA and QSA leads to solutions which better fulfil a query compared to retrieval with SIM. Further, QSA outperforms GA in this respect.*

Experimental Setup

We constructed a case base of 58 recipe workflows for pasta dishes from a real cooking web site[2]. For evaluation, we use compositional adaptation (Müller and Bergmann 2014), which decomposes each workflow within the case base into meaningful subcomponents, called workflow streams (e.g. subprocess to prepare the pasta sauce). During adaptation, deficiencies in the retrieved case are compensated by replacing fragments of the retrieved workflow by appropriate workflow streams (e.g. replacing the pasta sauce by a sauce from another workflow). From the 58 workflows 212 distinct workflow streams could be automatically extracted, which were stored in the adaptation knowledge repository. This setting constitutes the experimental condition (A) with a large amount of adaptation knowledge. Further, we prepared a second setting (B) in which the adaptation knowledge learned form 20 randomly selected cases is removed. In addition the adaptation method is tweaked in a way that these 20 cases are not adaptable any more.

We evaluated our retrieval method following the leave-one-out evaluation principle. For this purpose, we first compute a query based on each workflow from the case base as follows: for each workflow W_i we retrieved the most similar workflow W_j from the case base. We then compute a POQL query q_i by determining the set of nodes to be added (desired nodes) and deleted (undesired nodes) in order turn W_j into W_i (independent from any adaptation knowledge). Then retrieval and adaptation is performed for q_i while case W_i is temporarily removed from the case base as well as from the adaptability computation. In particular, W_i is removed from the tables of all cases in QSA and it is neglected in the computation of the global adaptability in GA. The adaptation knowledge, however, is the same for all queries. This avoids expensive re-learning of adaptation knowledge and prevents undesired effects caused by its variation. Please note that not the capability of adaptation is subject of the evaluation but its assessment. In the experiment we evaluated the performance of GA, QSA and SIM in both conditions (A) and (B). We measured for each query the gain in similarity that could be achieved by adaptation (to assess Hypothesis 1) as well as the final similarity of the resulting workflow with respect to the query (to assess Hypothesis 2).

[1] `cake.wi2.uni-trier.de`

[2] `www.studentrecipes.com/recipes/pasta/`

Condition	Similarity Gain by Adaptation			Solution Similarity		
	SIM	GA	QSA	SIM	GA	QSA
A	0.030	**0.076**	**0.149**	0.834	**0.850**	**0.850**
B	0.013	**0.063**	**0.123**	0.817	0.826	**0.832**

Table 2: Summary of experimental results

Results

The results of our experiment are summarized in Table 2. For each condition the table shows the similarity gain achieved by adaptation after retrieving the cases with SIM, GA, and QSA, respectively. The figures show the average values over all 58 queries. Under both conditions we can clearly see an improvement of GA over SIM as well as of QSA over SIM. Also QSA outperforms GA. All differences in the table are statistically significant as determined by a paired t-test ($p < 0.005$). Thus Hypothesis 1 is clearly approved. The table also shows the final solution similarity of the adapted case to the query. Under condition A, GA and QSA outperform SIM and QSA outperforms GA, as expected. All results are significant ($p < 0.05$). The results of GA and QSA are on a similar level, thus there is no advantage for QSA. Under condition B, with a smaller amount of adaptation knowledge, there is still an improvement of GA and QSA over SIM, but the improvement is clearly weaker and not statistically significant for GA over SIM. Except for this fact, Hypothesis 2 is approved.

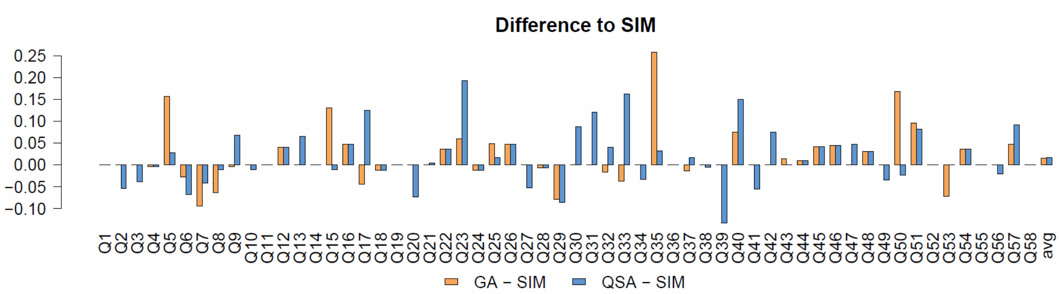

Figure 3: Difference in similarity of solution to the query, between GA and SIM as well as QSA and SIM under condition A

Figure 3 shows some more detailed results on the overall solution improvement per query for condition A. For a majority of queries we can see an improvement, but there are some exceptional queries (e.g. Q29) for which GA and/or QSA lead to worse results. For those queries, the assessment of the adaptability of the selected cases is quite high, while the actual similarity gain that could be achieved is low. Due to the wrong estimation, a case is retrieved which in total lead to a worse result.

Concerning the retrieval time, the GA approach only led to a slight increase. The average retrieval time per query for SIM over both conditions is 1.3 seconds and for GA it is 1.6 seconds. For QSA the retrieval time drastically increases to 85 (cond. A) and 115 (cond. A) seconds.

Conclusion

We introduced a novel approach for retrieving adaptable cases in POCBR following the idea of Leake et al. (1997) to learn the adaptability from the case base. Similar to Smyth and Keane's (1995) proposal to estimate case coverage from the case base, we determine the adaptability of cases. The transfer and detailed investigation of theses ideas in POCBR is the major novel contribution of this paper.

Our evaluation showed that GA and QSA lead to statistically significant improvements. However, the degree of similarity improvement is smaller than expected. This is partly caused by an overestimation of the adaptability. In addition it is a consequence of the fact that cases are widely adaptable by the employed compositional adaptation, which limits the benefit of improved retrieval. Although our results showed that QSA performs better than GA with respect to similarity improvement, it results in a largely increased retrieval time. A practical alternative could be an ensemble approach, computing the results from SIM and GA and selecting the best adapted case. This would only double the overall reasoning time, but as an additional analysis showed, it would clearly outperform GA and QSA under both conditions.

In the future we plan to develop an alternative approach for QSA, for example, by learning a model from the local adaptability table, which can be applied during retrieval, thus significantly shortening the retrieval time compared to QSA. Future research will also comprise an extended evaluation within other domains and using other POCBR adaptation methods such as adaptation operators or generalized cases (Müller and Bergmann 2015b; 2015a). Further, an extension towards the full expressiveness of POQL is planned. We also aim at investigating the impact of the amount of available adaptation knowledge in more detail.

Acknowledgments

This work was funded by the German Research Foundation (DFG), project number BE 1373/3-1.

References

Aamodt, A., and Plaza, E. 1994. Case-based reasoning: Foundational issues, methodological variations, and system approaches. *AI Communications* 7(1):39–59.

Bergmann, R., and Gil, Y. 2014. Similarity assessment and efficient retrieval of semantic workflows. *Inf. Syst.* 40:115–127.

Bergmann, R.; Richter, M. M.; Schmitt, S.; Stahl, A.; and Vollrath, I. 2001. Utility-Oriented Matching: A New Research Direction for Case-Based Reasoning. In *9th German Workshop on Case-Based Reasoning (GWCBR'2001).*

Bergmann, R.; Gessinger, S.; Görg, S.; and Müller, G. 2014. The collaborative agile knowledge engine CAKE. In *Proceedings of the 18th International Conference on Supporting Group Work*, GROUP '14, 281–284. ACM.

Bergmann, R. 1998. On the use of taxonomies for representing case features and local similarity measures. In *Proceedings of the 6th German Workshop on Case-Based Reasoning,GWCBR-98.*

Craw, S.; Wiratunga, N.; and Rowe, R. C. 2006. Learning adaptation knowledge to improve case-based reasoning. *Artificial Intelligence* 170(16):1175–1192.

Hanney, K., and Keane, M. T. 1997. The adaption knowledge bottleneck: How to ease it by learning from cases. In *Case-Based Reasoning Research and Development, Second International Conference, ICCBR-97*, volume 1266 of *LNCS*, 359–370. Springer.

Leake, D. B.; Kinley, A.; and Wilson, D. C. 1997. Case-based similarity assessment: Estimating adaptability from experience. In *AAAI/IAAI*, 674–679.

Lopez de Mantaras, R.; McSherry, D.; Bridge, D.; Leake, D.; Smyth, B.; Craw, S.; Faltings, B.; Maher, M.-L.; Cox, M.; Forbus, K.; Keane, M. T.; Aamodt, A.; and Watson, I. 2005. Retrieval, reuse, revision and retention in case-based reasoning. *The Knowledge Engineering Review* 20:215–240.

Minor, M.; Montani, S.; and Recio-Garca, J. A. 2014. Process-oriented case-based reasoning. *Information Systems* 40(0):103 – 105.

Müller, G., and Bergmann, R. 2014. Workflow Streams: A Means for Compositional Adaptation in Process-Oriented CBR. In *Case-Based Reasoning Research and Development, ICCBR-14*, volume 8765 of *LNCS*. Springer. 315–329.

Müller, G., and Bergmann, R. 2015a. Generalization of Workflows in Process-Oriented Case-Based Reasoning. In *28th FLAIRS Conference*. Hollywood (Florida), USA: AAAI.

Müller, G., and Bergmann, R. 2015b. Learning and Applying Adaptation Operators in Process-Oriented Case-Based Reasoning. In *Case-Based Reasoning Research and Development - 23nd International Conference, ICCBR 2015*, volume 9343 of *LNCS*, 259–274.

Müller, G., and Bergmann, R. 2015c. POQL: A New Query Language for Process-Oriented Case-Based Reasoning. In *Proceedings of the 17th LWA Workshops*.

Smyth, B., and Keane, M. T. 1994. Retrieving adaptable cases. In Wess, S.; Althoff, K.-D.; and Richter, M., eds., *Topics in Case-Based Reasoning*, volume 837 of *Lecture Notes in Computer Science*. Springer Berlin Heidelberg. 209–220.

Smyth, B., and Keane, M. T. 1995. Remembering to forget. In *Proceedings of the 14th international joint conference on Artificial intelligence*, 377–382. Citeseer.

Stahl, A. 2004. *Learning of Knowledge-Intensive Similarity Measures in Case-Based Reasoning*. Phd thesis, Technical University of Kaiserslautern.