A Case Study of Case-Based CBR*

David B. Leake, Andrew Kinley, and David Wilson

Computer Science Department
Lindley Hall 215, Indiana University
Bloomington, IN 47405, U.S.A.
{leake,akinley,davwils}@cs.indiana.edu

Abstract. Case-based reasoning depends on multiple knowledge sources beyond the case library, including knowledge about case adaptation and criteria for similarity assessment. Because hand coding this knowledge accounts for a large part of the knowledge acquisition burden for developing CBR systems, it is appealing to acquire it by learning, and CBR is a promising learning method to apply. This observation suggests developing *case-based* CBR systems, CBR systems whose components themselves use CBR. However, despite early interest in case-based approaches to CBR, this method has received comparatively little attention. Open questions include how case-based components of a CBR system should be designed, the amount of knowledge acquisition effort they require, and their effectiveness. This paper investigates these questions through a case study of issues addressed, methods used, and results achieved by a case-based planning system that uses CBR to guide its case adaptation and similarity assessment. The paper discusses design considerations and presents empirical results that support the usefulness of case-based CBR, that point to potential problems and tradeoffs, and that directly demonstrate the overlapping roles of different CBR knowledge sources. The paper closes with general lessons about case-based CBR and areas for future research.

1 Introduction

The role and relationship of multiple knowledge sources in case-based reasoning is receiving increasing attention from the CBR community. As pointed out by Richter (1995), the fact that CBR provides multiple overlapping "knowledge containers"—such as cases, similarity criteria, and case adaptation information—facilitates the development of CBR systems by enabling system developers to place knowledge in whichever container is most convenient. In addition, these multiple knowledge sources provide many opportunities for learning (e.g., Aha & Wettschereck, 1997).

Investigators have studied a range of analytic and inductive learning methods for refining the knowledge sources within CBR. For example, Hammond's (1989)

* This work was supported in part by the National Science Foundation under Grant No. IRI-9409348.

CHEF uses explanation-based methods to learn *ingredient critics* for use in adaptation, while Hanney and Keane (1997) and Wilke *et al.* (1997) propose inductive generalization to learn adaptation rules; Veloso's (1994) Prodigy/Analogy uses explanation-based methods to learn similarity criteria, while Ricci and Avesani (1995) advocate reinforcement learning. However, despite early work on using CBR within CBR systems, such as Sycara's (1988) study of case-based case adaptation, there has been little recent attention to such approaches. Yet CBR's advantages for top-level reasoning—ease of knowledge acquisition, ability to perform successfully despite imperfect domain theories, and simple learning—also suggest its potential benefit within CBR systems.

We refer to CBR performed by case-based components as "case-based CBR." This paper examines case-based CBR through a case study of the CBR system DIAL (Leake *et al.*, 1996), which uses CBR for both case adaptation and similarity assessment. DIAL's case-based adaptation was developed to address the classic knowledge acquisition problem for case adaptation. Preliminary studies showed that learning adaptation cases improved adaptation performance (Leake *et al.*, 1996), but the method also raised questions about how to refine similarity assessment as adaptation cases are acquired. The difficulty is that useful similarity judgments must reflect "adaptability" (Birnbaum *et al.*, 1991; Smyth and Keane, 1996). In a case-based adaptation system, adaptability is not static—it changes as adaptation cases are learned. Consequently, similarity judgments much change as well. This led us to investigate extending our system's internal case-based methods to use adaptation cases for similarity assessment as well, tying similarity judgments directly to the system's adaptation knowledge.

Case-based CBR raises a number of questions about the practicality of using case-based components in a CBR system:

1. **System design:** How should the components' knowledge be represented and organized?
2. **Knowledge acquisition:** How much specialized knowledge must be provided to support component CBR processes, and how does this effort compare to hand coding rules for these processes?
3. **Net efficiency:** How will use of case-based components affect the overall efficiency of the top-level CBR system?
4. **Learning interactions:** How do the effects of learning in the main case library and by the case-based components contribute individually to overall performance of a CBR system, and how do the multiple forms of learning interact?
5. **Coverage:** How does external feedback to the component CBR processes (e.g., during case adaptation) affect the range of problems that the top-level CBR system can solve?
6. **Utility of learning:** How does the proliferation of stored cases for the component CBR processes affect overall performance?

This paper examines these questions, discussing strategies, lessons, and issues arising from experience developing the case-based CBR system DIAL (Leake *et al.* 1996; 1997b). It begins with a synopsis of how basic CBR issues are addressed within DIAL's component CBR processes. It then highlights aspects of their performance and it discusses key lessons about the role and potential of case-based CBR.

2 Synopsis of DIAL

DIAL is a case-based planner in the disaster response planning domain. The system's task is to generate plans to guide damage assessment, evacuations, etc., in response to natural and man-made disasters such as earthquakes and chemical spills. DIAL's top-level planning component is based in a straightforward way on traditional case-based planning systems such as CHEF (Hammond, 1989). Given a new disaster, the response plan for a similar disaster is retrieved. The applicability of that plan to new circumstances is then evaluated by a simple evaluation component using the stereotype-based problem-detection process described in (Leake, 1992), with backup evaluation by a human user, in order to identify problems requiring adaptation. When problems are found, the plan and a description of the problem in a pre-defined problem vocabulary (either generated by the system or input by the user) are provided to the adaptation component. DIAL uses CBR both for similarity assessment during plan retrieval and for adaptation of the plans it retrieves. We describe these processes below.

3 A case-based framework for adaptation learning

DIAL's adaptation component begins with general domain-independent adaptation knowledge. It uses this knowledge to build up a library of adaptation cases to facilitate future adaptation of similar problems. DIAL's initial adaptation knowledge is a small set of abstract transformation rules and a library of "weak methods" for memory search, such as the "local search" strategy to find related concepts by considering nearby nodes in memory (Kolodner, 1993). Its adaptation rules are indexed under elements of a problem-description vocabulary. For example, the problem type FILLER-PROBLEM:UNAVAILABLE-FILLER indexes the transformation to substitute a role-filler. When presented with a new adaptation problem, DIAL first retrieves a transformation rule associated with the problem type. Each association between a problem type and transformation has associated information about how to determine—from the problem type—the parts of the plan to be transformed. The association also contains information about how to determine the information needed to apply the transformation. For substitutions within a schema, the system extracts from the schema constraints on the role whose filler is being substituted. After constraints have been identified and used to generate a knowledge goal (Hunter, 1990; Ram, 1987) for the information needed, it searches memory for that information.

The goal of adaptation learning is to learn the memory search strategies that apply to particular types of problems, in order to reuse them. Initially, however, the system has no memory search cases and must rely on weak methods such as local search. As a simple example, if part of a retrieved disaster response plan is to have the Red Cross deliver supplies, but there is no Red Cross in the country where a new disaster occured, a possible memory search path for a substitution would start at the memory node for Red Cross, move to its abstraction *relief organization*, and then move to specifications of that node (e.g., the Red Crescent). The memory search process is continued until it yields an acceptable result or reaches a limit on memory search effort. When DIAL is unable to generate an acceptable adaptation, an interactive interface allows a human user to guide it along the steps leading to a successful adaptation.

Once a successful adaptation has been generated, either by DIAL or by a human user, the system saves a trace of the steps used in its memory search process, packaged with the transformation rule used, as an *adaptation case* for future reuse. Learned adaptation cases make useful memory search paths explicitly available and may enable the system to solve adaptation problems that would otherwise have been impossible to solve within system resource limits.

4 Design choices and motivations

Designing a case-based component for a CBR system, like designing any case-based reasoning system, requires determining the type of CBR process to use (transformational or derivational), the case representation, the case organization scheme, and how to perform case adaptation. DIAL's case-based adaptation process reflects the following design decisions:

Transformational vs. derivational CBR: A key question for case-based case adaptation is the type of information that adaptation cases should store. Previous case-based adaptation systems store the solution of a prior adaptation—the change that was selected—and reapply it by transformational analogy (e.g., (Sycara, 1988)). This is appropriate when the derivation of the prior adaptation is not available. However, when derivations are available, derivational CBR is a natural means for providing flexible reuse (Veloso, 1994). Because DIAL's adaptation component has the trace of each new adaptation, it can use derivational analogy for case adaptation, even though DIAL must use transformational analogy for its top-level planning task, due to the lack of derivational information for the disaster response plans in its case library.

Case representation: DIAL's adaptation cases characterize adaptations by two types of information: general domain-independent transformations (e.g., substitute, add, delete) and the memory search information needed to apply them. This division is modeled on Kass's *adaptation strategies* (Kass, 1990), which have been shown capable of capturing a wide range of adaptations. However, the memory search procedures used by adaptation strategies are hand coded; DIAL's approach builds up memory search strategies from experience.

Case organization: DIAL's adaptation cases are organized by the problems they address, using a vocabulary of problem types similar to those that guide adaptation in numerous other CBR systems (e.g., Hammond, 1989; Leake, 1992). For example, if a candidate response plan is inappropriate because a role-filler is unavailable (e.g., a police commissioner may be out of town and unable to be reached in an emergency situation), the problem is described by the problem type FILLER-PROBLEM:UNAVAILABLE-FILLER, and that description is used as an index to retrieve adaptation cases for similar problems.

Adaptation: Using CBR to guide case adaptation prompts concerns about where the process will "bottom out." The need to adapt adaptation cases presents a new adaptation problem, and there is no reason to expect that developing adaptation rules for adaptation cases will be any easier than developing adaptation rules for the top-level system. DIAL's response is to strongly restrict adaptation of adaptation cases, by relying on a single domain-independent adaptation method. When following the memory search path in an adaptation case fails to identify a usable solution, DIAL seeks similar solutions by local search, starting from the end of the path. If that fails, it backtracks along the replayed memory search path, using local search from the points along that path. This gradually relaxes restrictions in the search path to find alternatives.

5 A case-based framework for similarity learning

Previous work on refining similarity criteria to match adaptation abilities focuses primarily on adjusting a set of similarity criteria (that approximate adaptability) rather than judging adaptability directly from adaptation knowledge (Birnbaum *et al.*, 1991). Research by Smyth and Keane (1996) takes a valuable step in replacing semantic similarity with adaptability, but does not address learning: Their method assumes adaptation knowledge is static and depends on adaptation rules being annotated with estimated costs by the system designer. In addition, their method uses a single estimate for an entire problem class. Tests show that this may not be sufficiently fine-grained (Leake *et al.*, 1997a).

A natural alternative is to use CBR for adaptability: To predict the adaptability of a problem from experience adapting similar problems. This approach enables similarity judgments to keep pace with learned adaptation experience and to provide fine-grained estimates of adaptation costs. Given a new disaster situation and a candidate response plan with applicability problems, DIAL's similarity assessment component retrieves the adaptation cases DIAL would apply to adapt each problem in the plan, and estimates the total cost of applying all the adaptation cases. Retrieved adaptation cases for the best plan are passed on to the adaptation component, in the spirit of Smyth and Keane (1996).

Ideally, in similar future contexts, replaying the same adaptation derivation will lead to a result that applies to the new context, so the length of the stored derivation is a good predictor of the re-application cost. However, differences between the old and new problems may prevent the prior derivation from being directly applicable, increasing the cost of adaptation. Consequently, DIAL multiplies the prior cost by a "dissimilarity" factor based on the semantic similarity of the old and new situations. This factor is simply the sum of distances between memory nodes for corresponding role-fillers in the problem descriptions, in the system's memory hierarchy. The rationale for this approach is that the guidance from an adaptation case is most useful when reapplied to closely-matching adaptation problems. Leake et al. (1997a) show that for a set of test problems in DIAL's domain, this method retrieves more adaptable cases than either standard similarity assessment methods or methods based on average adaptation costs for classes of problem types.

6 Design choices and motivations

A central question for developing a case-based similarity assessment component is what constitutes "similarity." Other key issues are the case representation, indexing, and adaptation.

Relating similarity to adaptability: Traditional similarity assessment uses semantic similarity as a proxy for adaptability; more recent approaches, led by Smyth and Keane's adaptation-guided retrieval, call for replacing semantic similarity with adaptability. DIAL's approach to similarity assessment emphasizes the importance of adaptability, but its approach to assessing adaptability is tempered by the principle that *minimizing total solution generation time takes precedence over minimizing adaptation time*. This principle was examined for the top-level CBR process by Veloso and Carbonell (1991), and is crucial to internal CBR as well.

To perform a low-cost initial filtering of candidate response plan cases, DIAL uses semantic similarity between the current and prior situations. After initial filtering, finer-grained filtering is based on the seriousness of differences (adaptability), rather than the level of semantic similarity. We are now investigating the tradeoffs resulting from different levels of effort in judging adaptability.

Other key issues: Because DIAL's case-based similarity component relies on the same case library as its case-based adaptation, its approach to other design issues closely follows the design choices for case-based adaptation. The case representation required for adaptation cases to support similarity assessment is virtually unchanged from that required for adaptation: The only added information is a count of the number of steps performed in the adaptation, which is stored for efficiency but could be re-calculated from the derivational trace. The indexing criteria used to retrieve adaptation cases when estimating adaptation costs are the same ones initially developed to retrieve adaptation cases when performing adaptation after similarity-based plan retrieval. Adaptation of adaptation cases during similarity assessment is simple: When adaptation cases are used to estimate the adaptation cost of new problems, the new cost is estimated by multiplying the old cost by the dissimilarity factor.

7 Lessons Learned

The previous sections describe DIAL's case-based adaptation and similarity assessment. Tests of DIAL's performance with these methods provide a first set of data points illuminating our general questions about the practicality and performance of using case-based CBR. We will address each question in turn, first discussing knowledge acquisition and efficiency issues, then examining how learning from the component CBR processes affects the range of problems the system can solve, and finally considering the potential utility problems accompanying case-based CBR. We view the primary interest of these results not as being specific to DIAL, but as a demonstration that case-based CBR can be practical and as an illustration of issues to address.

Knowledge acquisition: The knowledge requirements for DIAL's case-based adaptation fall between those for knowledge-intensive explanation-based methods and pure inductive approaches. DIAL does not require the detailed knowledge usually coded into adaptation rules, but does rely on three other forms of knowledge in order to learn and reapply its adaptations. The first is a semantic network of known domain concepts. This is a standard part of many CBR systems, and is routinely used for domain-independent adaptation methods (e.g., to find substitutions by local search (Kolodner, 1993)). Although this network must obviously include the concepts important for a CBR system's task, DIAL's case-based adaptation process can learn to use it effectively even if the most relevant connections are not pre-coded. The second type of knowledge enabling DIAL's case-based adaptation is a categorization of types of abstract transformations. A number of researchers have argued that considerable coverage can be achieved by a small set of such transformations (Carbonell, 1983; Hinrichs, 1992). The third type of knowledge is a vocabulary of problem types requiring adaptation. Such vocabularies are widely used by CBR systems to organize adaptation knowledge, so they are a prerequisite not only to DIAL's case-based methods, but also to effective use of hand-coded adaptation rules. In addition, they appear to have a wide range of applicability, minimizing the need to develop multiple vocabularies for different tasks. Thus DIAL's case-based adaptation approach facilitates knowledge acquisition compared to rule-based approaches, by avoiding manual coding of rules and requiring only standard supporting knowledge. In addition, unlike pure inductive approaches, the case-based approach can learn new adaptations from single examples.

Net efficiency: As discussed in detail in Leake, Kinley, and Wilson (1997b), we examined efficiency in a set of trials generating response plans for 18 disasters, starting from a case library of 5 response plans and performing 118 adaptations. Conditions were no learning (NL), plan case learning (CL), adaptation case learning (AL), combined adaptation learning and plan case learning (using semantic similarity to retrieve plan cases) (AL+CL), and the combination of adaptation, plan, and similarity learning (AL+CL+SL).

Figure 1 shows the average total execution time per problem solved, separated into two parts, (1) time spent in retrieval and similarity assessment, and (2) time spent in adaptation. One surprising result was the large speedup provided by adaptation learning alone, compared to case learning alone. Even more surprising was that the addition of case learning, in the AL+CL condition, degraded performance compared to adaptation learning alone. Adding similarity learning (AL+CL+SL) restored the efficiency to equal that of AL. The significance of these results will be discussed in the next section. Retrieval times increase from

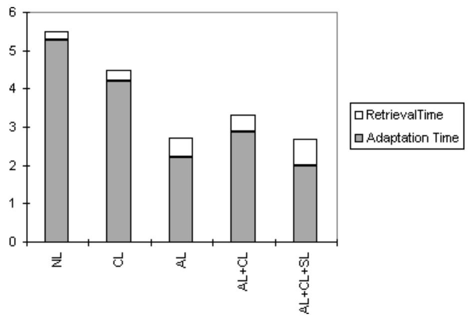

Fig. 1. Average retrieval/similarity assessment and adaptation time.

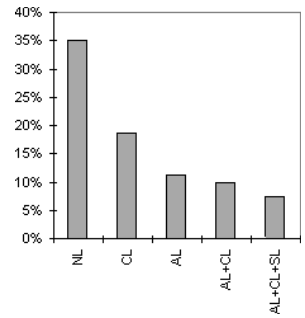

Fig. 2. Failure rates.

NL to CL (as expected from searching a larger plan case library), to AL (because the adaptation library to search through grows faster than the case library for CL, requiring more search effort for adaptations), and to AL+CL. The drop in retrieval time for AL+CL+SL is unexpected and merits investigation.

Learning interactions and efficiency: Given that CL and AL are each effective individually, we expected AL+CL to have better performance than either alone. However, adaptation cost with AL+CL increases compared to AL. Our explanation is that the cases that appear most relevant, according to static similarity criteria, may not be the easiest to adapt. The mismatch between static similarity criteria and learned adaptation abilities did not appear to cause significant problems with AL, however. Our explanation was that for AL, adaptations were being learned to apply to a small fixed library of plan cases, making it likely (after enough problems were processed) that any plan case chosen had been applied to a similar prior disaster, so the adaptations selected by similarity were generally appropriate. However, when novel plan cases are added to the case base by case learning (as in AL+CL), the newly-learned plans must initially be adapted with adaptation cases that were developed in other contexts and are not as directly applicable, increasing adaptation cost. AL+CL+SL increases retrieval cost but selects more adaptable cases, decreasing adaptation cost and resulting in similar total time.

Coverage and quality: In addition to efficiency, an important measure of the performance of a CBR system is the range of problems it can solve. DIAL's initial domain theory is incomplete, but its ability to store and reuse user-provided solutions (both disaster response plans and adaptations) allows it to augment its knowledge. In addition, the ability to reuse memory search paths from learned adaptation cases enables it to explore regions of memory that might otherwise have been too expensive to explore. Figure 2 shows the percentage of the trial problems the system could not find a satisfactory solution, for each of the learning conditions. Again, adaptation learning alone performed better than case learning alone, but the interference effect between adaptation and case learning

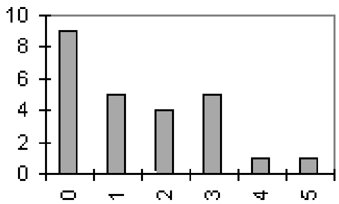

Fig. 3. Number of adaptation cases generated at each level of reuse.

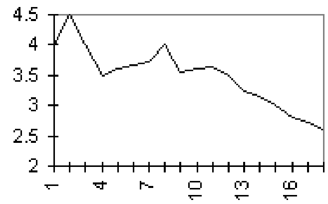

Fig. 4. Number of novel adaptations per plan as a function of number of plans processed.

(AL+CL) that degraded efficiency compared to AL alone did not affect coverage. AL+CL+SL provided slightly better coverage than the other methods in these tests.

Utility of learning: Because the previous tests were performed on a comparatively small set of problems, we have no specific data on how performance is affected as many cases are learned. However, retrieval costs are a potential problem as the library of internal cases grows. One possible response would be to reduce case library size through selective "forgetting" of adaptation cases (Smyth and Keane, 1995). In DIAL's domain, precise analysis of which cases to delete is not possible, and efficiency—which may be furthered by retaining some cases not necessary for competence—is important as well. However, simple forgetting strategies may be useful. Figure 3 shows the number of times each learned adaptation case is reused in a sample run; the x-axis is the level of reuse (e.g., "1" for cases reused only once), and the y-axis is the number of cases at that level of reuse (e.g., five learned adaptation cases were reused only once). On this set of trials, roughly 20% of the cases were never reused. Adaptation cases that are not reused, or are not reused sufficiently frequently, could be deleted.

Figure 4 graphs the average number of new adaptation cases created per response plan, processed as a function of the number of response plans generated. It shows a rapid decrease in the number of adaptation problems that require reasoning from scratch from processing a comparatively small number of adaptations. Thus it might also be possible, for example, to control the utility problem by simply stopping adaptation learning after establishing a set of adaptation cases expected to produce adequate coverage.

8 Lessons about case-based CBR

Our previously-described research led to the following observations about case-based CBR:

- **Adaptation learning can be as important as case learning:** Our sample runs showed that for a given set of problems, the efficiency and coverage improvements from learning new adaptation cases with a fixed case library can surpass those of case learning with fixed adaptation knowledge.

- **Simple domain-independent methods can be sufficient for internal CBR processes:** Even DIAL's simple methods for case-based adaptation and similarity learning, which require minimal knowledge acquisition, markedly improve its performance. For example, semantic similarity can provide a useful pre-filtering stage during retrieval.
- **The results of multiple learning processes must be used in a coordinated way:** In our tests, problem-solving efficiency with AL+CL was worse than with AL alone, apparently because case selection did not take learned adaptation knowledge into account, preventing the system from making the best possible use of new cases it had learned. Learned similarity assessment criteria in AL+CL+SL coordinated selection of learned cases to fit learned adaptations, enabling the system to choose the most adaptable learned case given its learned adaptation knowledge.
- **Sharing a single case library between case-based components is a convenient way to coordinate learning:** Cases storing traces of adaptations already provide sufficient information to estimate adaptability, enabling a single case library to serve for both adaptation and similarity assessment. This assures that knowledge for both processes is synchronized.
- **Proliferation of internal cases is a potential problem:** Cases for the component processes of a CBR system may be learned at a much higher rate than cases for the top-level CBR system, and the usefulness of additional learned cases may drop rapidly.

9 Future directions

The previous results show that case-based components within a CBR system can improve performance compared to case learning alone. However, since our observations are based on limited tests in a single domain, a pressing future need is to extend the study to multiple domains and larger problem sets.

Our research so far has focused on case-based methods for adaptation and similarity assessment. Another rich research area is control of the case retrieval process. It is appealing for a case memory to learn which types of queries it tends to receive, which classes of cases are relevant to them, and which strategies are appropriate to retrieve these cases. Case-based reasoning appears to be a promising method for this task. For example, if the case retrieval process is modeled as strategic memory search (e.g., following Kolodner 1984), it will be possible to apply results from existing research on case-based and introspective analogy for learning memory search (e.g., as in DIAL's adaptation cases and Kennedy, 1995). CBR for retrieval provides another opportunity for sharing case bases between multiple component CBR processes: both the memory search paths followed during adaptation and those followed during case retrieval can form a single case library of memory search paths for future use.

Using case-based methods for learning retrieval information also has ramifications for case-based case storage. Cases that describe where to find relevant information in memory also describe where related new information should be stored; conversely, cases describing where information has been stored also describe where related information should be found. By using a single case library of search paths to guide both case retrieval and storage, learned storage and retrieval knowledge can be coordinated. This example shows that key lessons from applying case-based methods to the similarity and adaptation components of a CBR system may be useful for other components as well.

10 Conclusion

This paper identifies central questions about case-based CBR and presents a case study demonstrating that it can be a practical method, in terms of knowledge acquisition, processing efficiency, and quality of solutions. The key result is to show that simple CBR methods can be practical for guiding components of a CBR system, for refining their knowledge sources, and for making coordinated use of the results of their learning. Our tests also support the value of adaptation learning: In some situations, learning only adaptations has even greater benefit than learning new problem cases. This provides—to our knowledge—the first empirical demonstration of the overlapping contributions of case and adaptation knowledge within CBR. Overall, a combination of adaptation, case, and similarity learning provided comparable efficiency and slightly better problem coverage than any of the other methods. These results emphasize the potential of case-based CBR, the complex interactions between different types of learning, and the need for further study of multiple learning processes within CBR systems.

References

[Aha and Wettschereck, 1997] D. Aha and D. Wettschereck. Case-based learning: Beyond classification of feature vectors. Call for papers of ECML-97 workshop, 1997.

[Birnbaum *et al.*, 1991] L. Birnbaum, G. Collins, M. Brand, M. Freed, B. Krulwich, and L. Pryor. A model-based approach to the construction of adaptive case-based planning systems. In R. Bareiss, editor, *Proceedings of the DARPA Case-Based Reasoning Workshop*, pages 215–224, San Mateo, 1991. Morgan Kaufmann.

[Carbonell, 1983] J. Carbonell. Learning by analogy: Formulating and generalizing plans from past experience. In R. Michalski, J. Carbonell, and T. Mitchell, editors, *Machine Learning: An Artificial Intelligence Approach*, pages 137–162. Tioga, Cambridge, MA, 1983.

[Hammond, 1989] K. Hammond. *Case-Based Planning: Viewing Planning as a Memory Task*. Academic Press, San Diego, 1989.

[Hanney and Keane, 1997] K. Hanney and M. Keane. The adaptation knowledge bottleneck: How to ease it by learning from cases. In *Proceedings of the Second International Conference on Case-Based Reasoning*, Berlin, 1997. Springer Verlag.

[Hinrichs, 1992] T. Hinrichs. *Problem Solving in Open Worlds: A Case Study in Design*. Lawrence Erlbaum, Hillsdale, NJ, 1992.

[Hunter, 1990] L. Hunter. Planning to learn. In *Proceedings of the Twelfth Annual Conference of the Cognitive Science Society*, pages 261–268, Cambridge, MA, July 1990. Cognitive Science Society.

[Kass, 1990] A. Kass. *Developing Creative Hypotheses by Adapting Explanations.* PhD thesis, Yale University, 1990. Northwestern University Institute for the Learning Sciences, Technical Report 6.

[Kennedy, 1995] A. Kennedy. Using a domain-independent introspection mechanism to improve memory search. In *Proceedings of the 1995 AAAI Spring Symposium on Representing Mental States and Mechanisms*, pages 72–78, Stanford, CA, March 1995. AAAI Press. Technical Report WS-95-05.

[Kolodner, 1984] J. Kolodner. *Retrieval and Organizational Strategies in Conceptual Memory.* Lawrence Erlbaum, Hillsdale, NJ, 1984.

[Kolodner, 1993] J. Kolodner. *Case-Based Reasoning.* Morgan Kaufmann, San Mateo, CA, 1993.

[Leake *et al.*, 1996] D. Leake, A. Kinley, and D. Wilson. Acquiring case adaptation knowledge: A hybrid approach. In *Proceedings of the Thirteenth National Conference on Artificial Intelligence*, pages 684–689, Menlo Park, CA, 1996. AAAI Press.

[Leake *et al.*, 1997a] D. Leake, A. Kinley, and D. Wilson. Case-based similarity assessment: Estimating adaptability from experience. In *Proceedings of the Fourteenth National Conference on Artificial Intelligence.* AAAI Press, 1997.

[Leake *et al.*, 1997b] D. Leake, A. Kinley, and D. Wilson. Learning to integrate multiple knowledge sources for case-based reasoning. In *Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence.* Morgan Kaufmann, 1997. In press.

[Leake, 1992] D. Leake. *Evaluating Explanations: A Content Theory.* Lawrence Erlbaum, Hillsdale, NJ, 1992.

[Ram, 1987] Ashwin Ram. AQUA: Asking questions and understanding answers. In *Proceedings of the Sixth Annual National Conference on Artificial Intelligence*, pages 312–316, Seattle, WA, July 1987. Morgan Kaufmann.

[Ricci and Avesani, 1995] F. Ricci and P. Avesani. Learning a local similarity metric for case-based reasoning. In *Proceedings of the First International Conference on Case-Based Reasoning*, pages 301–312, Berlin, October 1995. Springer Verlag.

[Richter, 1995] Michael Richter. The knowledge contained in similarity measures. Invited talk, the First International Conference on Case-Based Reasoning, Sesimbra, Portugal., October 1995.

[Smyth and Keane, 1995] B. Smyth and M. Keane. Remembering to forget: A competence-preserving case deletion policy for case-based reasoning systems. In *Proceedings of the Thirteenth International Joint Conference on Artificial Intelligence*, pages 377–382, Montreal, August 1995. IJCAI.

[Smyth and Keane, 1996] B. Smyth and M. Keane. Design à la Déjà Vu: Reducing the adaptation overhead. In D. Leake, editor, *Case-Based Reasoning: Experiences, Lessons, and Future Directions.* AAAI Press, Menlo Park, CA, 1996.

[Sycara, 1988] K. Sycara. Using case-based reasoning for plan adaptation and repair. In J. Kolodner, editor, *Proceedings of the DARPA Case-Based Reasoning Workshop*, pages 425–434, San Mateo, CA, 1988. Morgan Kaufmann.

[Veloso and Carbonell, 1991] M. Veloso and J. Carbonell. Variable-precision case retrieval in analogical problem solving. In R. Bareiss, editor, *Proceedings of the DARPA Case-Based Reasoning Workshop*, pages 93–106, San Mateo, 1991. Morgan Kaufmann.

[Veloso, 1994] M. Veloso. *Planning and Learning by Analogical Reasoning.* Springer Verlag, Berlin, 1994.

[Wilke *et al.*, 1997] W. Wilke, I. Vollrath, K.-D. Althoff, and R. Bergmann. A framework for learning adaptation knowedge based on knowledge light approaches. In *Proceedings of the Fifth German Workshop on Case-Based Reasoning*, 1997.

This paper appears in *Case-Based Reasoning Research and Development: Second International Conference on Case-Based Reasoning*. Springer Verlag, Berlin, 1997, pp. 371-382.