Proceedings of the Thirteenth International Joint Conference on Artificial Intelligence, 1995.

Using Introspective Reasoning to Refine Indexing*

Susan Fox and **David B. Leake**
Computer Science Department
Lindley Hall 215
Indiana University
Bloomington, IN 47405 USA
E-mail: {sfox,leake}@cs.indiana.edu

Abstract

Introspective reasoning about a system's own reasoning processes can form the basis for learning to refine those reasoning processes. The ROBBIE[1] system uses introspective reasoning to monitor the retrieval process of a case-based planner to detect retrieval of inappropriate cases. When retrieval problems are detected, the source of the problems is explained and the explanations are used to determine new indices to use during future case retrieval. The goal of ROBBIE's learning is to increase its ability to focus retrieval on relevant cases, with the aim of simultaneously decreasing the number of candidates to consider and increasing the likelihood that the system will be able to successfully adapt the retrieved cases to fit the current situation. We evaluate the benefits of the approach in light of empirical results examining the effects of index learning in the ROBBIE system.

1 Introduction

A number of studies have examined the use of metareasoning to control the application of system domain knowledge and to guide acquisition of domain knowledge (e.g., [Bradzil and Konolige, 1990; Davis, 1982]). A more recent use of introspective reasoning is to monitor a system's own reasoning processes in order to refine those processes by failure-driven learning (e.g., [Collins *et al.*, 1993; Ram and Cox, 1994; Cox and Freed, 1995]). This paper presents an approach to introspective reasoning for refining the case retrieval criteria of a case-based planning system.

In the approach we are investigating, an introspective reasoning component monitors the processing of a case-based reasoning system and evaluates that processing in comparison to expectations for the ideal performance of the case-based reasoning process (for example, the expectation that the case retrieved will be the one that is easiest to adapt to the new situation). Failures are detected and learning occurs when performance deviates from the ideal. The failure is explained, an appropriate repair is identified, and the underlying system is modified to correct the failure. We have developed a system that applies this process to refine the indices used for case retrieval in a case-based planning system. Our program, ROBBIE, combines a case-based planner as its performance system with an introspective reasoning component which monitors its reasoning and guides introspective learning [Fox and Leake, 1995; 1994].

Applying introspective reasoning to CBR is novel in that learning in CBR systems is generally focused on acquiring domain knowledge, by acquiring new cases. ROBBIE learns new cases, but its primary focus is on the use of introspective reasoning to refine how its cases are applied. The component described in this paper, which refines indexing, is fully implemented. Our ultimate goal is to apply introspective learning to all parts of the CBR process—retrieval, adaptation, and evaluation of cases.

This paper has two parts. The first describes the ROBBIE system and its approach to introspective learning. The second presents an evaluation of the effects of introspective reasoning on ROBBIE's learning process. The desired effect is an improved ability to focus on relevant cases: a decreased average number of candidate cases retrieved, and the ability to retrieve cases that the system is more likely to adapt successfully. We evaluate system performance in light of these goals and briefly sketch the new costs that introspective reasoning entails and how the benefits balance against them. We close by relating our approach to other work on re-indexing and learning at a meta-level, and describing future directions for the ROBBIE project.

2 Motivations

Case-based reasoning systems use stored past experiences to solve current problems, and learn by storing new solutions for future use [Kolodner, 1993; Riesbeck and Schank, 1989]. When a new problem is presented, its description is used to create indices to guide retrieval; the cases in memory with the most similar indices are retrieved. The retrieved cases are adapted to fit the new situation and the resulting solution is stored in memory

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

[1] Re-Organization of Behavior By Introspective Evaluation

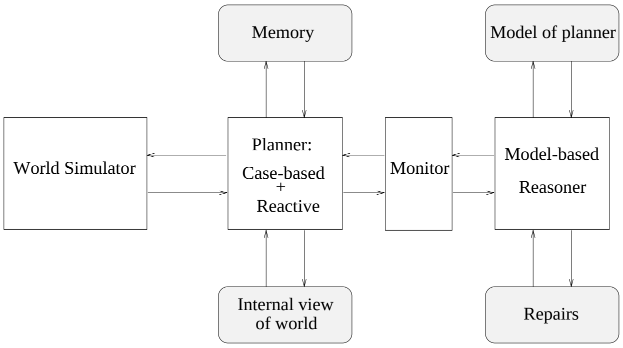

Figure 1: ROBBIE: Combining CBR with Introspective Reasoning

A	*Adaptation will finish in less than "max-steps-taken" number of steps*
B	*Similarity assessment will rank cases correctly* (the truth of this assertion is defined by the truth of three specific assertions it is linked to)
C	*The retrieved case will be correctly formed* (this also depends on more specific assertions for its truth, including D)
D	*The retrieved case will contain some plan steps* (this communicates its truth up to C: the value is found by examining underlying knowledge structures)

Table 1: Assertions from the ROBBIE model

for use under similar circumstances in the future.

The performance of any CBR system depends on using the right features for indexing cases, and having a measure of similarity that correctly selects adaptable cases from memory. If the wrong case is retrieved, adaptation may be costly or even impossible, depending on the CBR system's adaptation capabilities. The problem of determining feature relevance and good measures of similarity for retrieving cases may be difficult and time-consuming for the system designer.

One aim in combining an introspective reasoner with a case-based reasoning system is to allow the CBR system itself to determine when and how to refine its indexing criteria. ROBBIE can improve its assessment of similarity over time by learning the correct features to use in indexing, and extensions are underway to enable it to improve other parts of the CBR process as well (such as case adaptation).

3 The ROBBIE System

ROBBIE has two main parts, a case-based planning system that develops plans for traveling through an unfamiliar city (and applies them through simulated execution), and an introspective model-based reasoner, as shown in Figure 1. The planner carries out its low-level task and interacts with the simulated world, and in parallel the introspective reasoner monitors the planner's reasoning processes. If a problem is detected, the introspective reasoner may suspend planning while it repairs the reasoning problem, or it may permit the planner to continue and wait until more information becomes available to make a repair.

3.1 The Performance Task: Pedestrian Navigation

The performance task of ROBBIE is to navigate around a simulated set of city streets as a pedestrian. ROBBIE is given a starting location and a goal location and must create a plan for getting from the one to the other. Because the system does not have a perfect map nor a perfect retrieval mechanism, generation of a candidate plan does not assure successful execution. After plans are generated, ROBBIE executes them, using a simulated robot based on Firby's reactive planner [Firby, 1989]. This plan execution serves as a means of evaluating the quality of the CBR-created plan, as well as permitting ROBBIE to arrive at a solution despite flaws in the CBR-produced plan, providing a way to augment its case base. In addition, the reactive planner handles facets of the simulated world that cannot be predicted by the CBR system, such as traffic lights and blocked streets.

The plans in memory are indexed by their starting and ending locations, as well as other features ROBBIE has learned to use through its introspective reasoning. Plan retrieval in ROBBIE is a two-stage process: first ROBBIE gathers a pool of generally similar candidate cases, then it selects the best candidate from that pool. Plans describe navigational actions in terms of high-level plan steps such as "Move north on Oak to Birch" or "Turn east on Fir." These steps are then used as sequential goals for reactive execution, which breaks the problem down into individual steps for the simulated robot to execute.

3.2 The Introspective Task

The higher level task for ROBBIE is to improve its reasoning processes in response to detected reasoning failures. ROBBIE's approach uses an explicit, declarative model describing the underlying system's reasoning processes [Fox and Leake, 1994; Birnbaum *et al.*, 1991; Freed and Collins, 1994]. The model provides expectations about the ideal reasoning behavior of the system; the actual reasoning of the system is compared to this ideal as a means of detecting reasoning failures. Once a reasoning failure is detected the model is used to create an explanation of the failure and to suggest a repair (See [Fox and Leake, 1994] for a discussion of ROBBIE's failure detection). Currently, ROBBIE can learn new features to use in indexing memory, and can re-index its memory to pay attention to those features.

The introspective model consists of a structured set of "assertions" about portions of the CBR reasoning process. Each assertion describes a feature that would be true of an ideal CBR system. Table 1 lists a few of the assertions from the model for ROBBIE. Assertions range in specificity from implementation details of ROBBIE, such as assertion A in table 1, to abstract descriptions of the CBR process and the flow of control between its

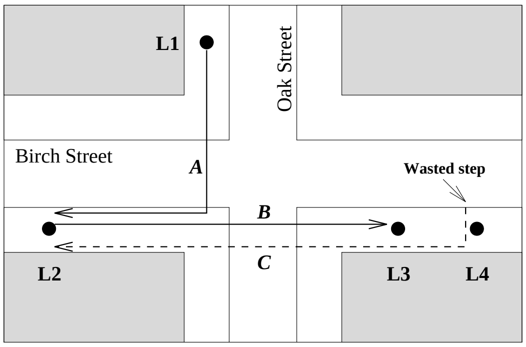

Figure 2: Implemented Example

components, such as assertions B and C.

The assertions are organized by the component to which an assertion refers, by the level of specificity of the assertion, and by connections to other related assertions. Dividing by component facilitates monitoring; the only assertions which must be monitored refer to the current component of processing. Assertions are arranged hierarchically to simplify the task of updating the model. Each assertion is linked to the other assertions which are related to it: as an abstraction, a specification, coming in sequence, or because a failure in one suggests a failure in the other. These links are used in explaining failures and finding repairs by focusing on the most fruitful areas of inquiry. Assertions C and D in table 1 show an abstract assertion and its specification. The vocabulary for representing assertions uses simple predicate calculus formulations. In [Fox and Leake, 1995] we provide a detailed description of the model and the representation it uses, and describe in more detail the need for a hierarchy of assertions.

After generating a successful solution, most CBR systems will simply store the solution and stop processing, while ROBBIE evaluates its retrieval process by retrieving the case in memory with the closest solution. If it is not the case that was retrieved originally, a retrieval failure occurred during the original retrieval. In repairing a retrieval failure ROBBIE must determine what, if any, relevant feature was overlooked. ROBBIE currently considers a pre-defined set of feature types such as "starts at location X", "ends on street Z", or "moves straight along street Y", which are ranked by a fixed feature hierarchy. ROBBIE uses the feature types to compare the new solution with the closest solution in memory, and explains the failure according to the missing features. In the future we plan to expand this mechanism to more general features using explanation-based generalization.

3.3 An Example

In this section we will present an implemented example illustrating the model of introspective reasoning we are developing and its benefits. Figure 2 shows the portion of the world map relevant to this example. The goal is to get from location L3 to location L2. ROBBIE has in memory plan A for going from L1 to L2 (Turn south, Move south to south side of Birch, Turn west, Move west to L2) and plan B for going from L2 to L3 (Turn east, Move east to L3). Using only starting and ending locations to judge similarity, plan A appears to be the closest because it shares the same ending location (ROBBIE's retrieval criteria do not include reversals of known routes). Plan A is selected and adapted to create plan C (dashed line in Figure 2): Turn south, Move south to south side of Birch, Turn west, Move west to L2.

During this process, the introspective reasoner monitors the system's behavior but detects nothing wrong. When the plan is executed, however, the wasted plan steps will be eliminated, producing a straight-line plan: Turn west, Move west to L2. When this resulting plan is stored into memory, a reasoning failure is detected: the final solution's steps are most similar to a plan (plan B) other than the retrieved case.

In explaining the cause of this failure, ROBBIE reconsiders related assertions in its model of the desired system behavior: that retrieval will operate successfully; that the index for retrieval will produce the closest case; that the index will include all the relevant features to retrieve the closest case. In re-testing the last assertion (in the context of a failure), the system discovers a feature of the case it had not used before: that each involves moving straight east or west. This failed assertion suggests a repair: add "moving east or west" to the features used in the index, and re-index all cases in memory to include the new feature.

4 Empirical Evaluation of Re-indexing

There has been little evaluation of the quantitative effects on performance from introspective reasoning; arguments for introspective reasoning have tended to focus on the capabilities provided in principle illustrated by particular examples. Although such work is important, we believe it is also important to define concrete measures which demonstrate the benefit of introspective reasoning, and weigh those against the costs of additional processing. For adding introspective reasoning to case-based reasoning, this requires examining whether isolated examples map onto a general trend of improvement beyond that accounted for by case acquisition alone.

We ran a series of tests on the ROBBIE system to evaluate introspective learning versus case learning alone.[2] We looked at two measures of ROBBIE's performance: the number of successfully completed plans at the end of a test run and the percentage of cases in memory considered for each retrieval. The number of successful plans provided a comparative measure of the level of success, and we predicted that introspective reasoning and re-indexing should enable ROBBIE to be more successful on a sequence than case learning alone. The percentage of plans considered during retrieval measured the efficiency of ROBBIE's processing over the course of a test run; we expected that the percentage should drop with re-indexing as new features allowed finer-grained distinctions between cases. In sum, we expected introspective

[2]ROBBIE was presented with a total of 41,600 goals: 26 sequences, each 40 goals long; and 20 runs with introspection, 20 without.

re-indexing to increase the success of ROBBIE *at the same time* as it decreased the work done in considering cases for retrieval.[3]

The experiments were designed to test performance from a knowledge-poor starting point, the most difficult situation for both case-based and introspective learning. ROBBIE started with an extremely small initial case base, spanning only three cases. The model used during testing contained a restricted set of assertions, to streamline the diagnosis process.

ROBBIE treats a sequence of locations as sequential goals: it starts at the first position, plans from there to the second, from the second to the third, and so on. For the purposes of these tests, when ROBBIE fails to reach a goal and is unable to recover, ROBBIE's location is set to the failed goal and the sequence continues from that point. When this occurs, no case is added to the case base: the size of the case base therefore matches the number of successes ROBBIE has.

We used a set of 40 randomly-created locations as goals for this experiment. These locations did not correspond exactly to locations in any plan in the initial case base. From these locations, we generated a set of 26 test sequences; each sequence was an ordering of the 40 locations with each location appearing exactly once. One sequence was designed by hand to extend the case base a little bit at a time: it started with goal locations similar to those in the case base, and gradually moved to goals further away from the initial cases. From the handmade sequence we created five groups of five sequences each, where each group varied from the handmade sequence by a different amount (25%, 50%, 75%, and 100% permuted; 25% permuted means 25% of the locations in the sequence were randomly swapped with other locations). In addition, we generated 5 completely random sequences to avoid any residual bias of the original sequence.

ROBBIE chooses at random between two candidate cases which appear equally similar. It is often the case, however, that the two cases produce different results which may affect later processing: one might fail or cause a new feature to be learned. To factor out these differing results, we ran each sequence twenty times with introspective re-indexing and twenty times without.

5 Empirical Results

5.1 Success Rate

We will first consider the success rate with and without re-indexing, defined by the number of successfully completed plans for a run. Figure 3 shows a breakdown of the results. The numbers for each sequence were averaged across the twenty runs; the averages for a particular sequence with and without re-indexing are side by side with the darker shaded bars indicating introspective re-indexing and the lighter shaded, no re-indexing. For all but one sequence, ROBBIE with introspective

[3]It would also be possible for introspective learning to cause retrieval to become too restrictive, decreasing success; verifying that this is not commonly the case is another reason for empirical tests.

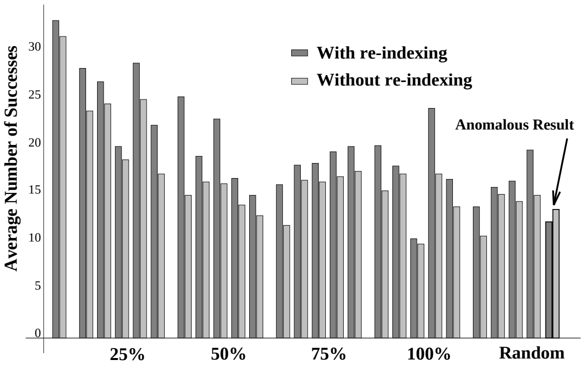

Figure 3: Average number of successes for each sequence; rightmost failed to improve with introspective re-indexing

learning did succeed more often than ROBBIE with only case learning, as we expected. This reflects the retrieval of more appropriate cases. The average of successful cases varied widely across the sequences, as did the difference between runs with introspective reasoning and those with case learning alone. In some cases, the difference between the two was very large; in others, very small. In one anomalous sequence, introspective reasoning performed slightly worse than case learning alone. That unexpected result is clearly troublesome, and we will return to discuss the anomaly after considering the second measure of ROBBIE's performance.

5.2 Improved Retrieval Efficiency

We will now consider the second measure of ROBBIE's performance, the extent to which learning new indices improves the efficiency of its retrieval. We measured this by considering what percentage of stored plans ROBBIE considered similar for each retrieval made during a test run. We graphed the course of the run on the x-axis as a time measure by counting each successive retrieval as the next point in time. The percentage of memory considered at each retrieval was graphed on the y-axis.

The effect of re-indexing can be seen in Figure 4, where we graphed a run with re-indexing alongside a representative sample of runs without it. The heavy line shows the percentage of cases considered in retrieval with introspective learning; each point where introspective learning took place is marked with an asterisk. The light lines are five runs without introspective learning. The runs without re-indexing varied from each other only in minor ways (which is why often only one line is visible). The introspective run began in the same way but the percentage of cases considered dropped sharply after learning the first new feature. While the percentage considered rose in some instances to match the runs without re-indexing, it returned to a consistently low level at other points. Early in the sequence, and again around retrieval 30, the trends in percentages considered mimicked the rise and fall of the runs without re-indexing, but at a much lower level. It is interesting to note that most of the learning took place at low points instead of peaks. This suggests that at the low points, retrieval might have been over-restricted, and new features are

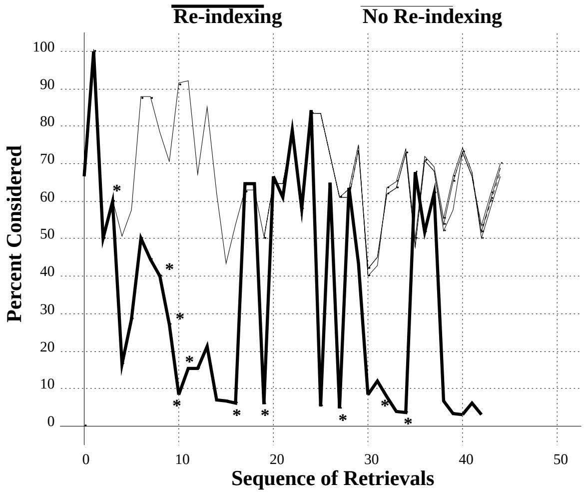

Figure 4: Learning new features caused the percentage considered to drop where non-learning runs continue at a high level (Handmade sequence)

learned to bring other cases into consideration. It also suggests that when no cases in memory are similar to a goal, large percentage retrievals are appropriate.

As is clear from Figure 4, the percentage considered from one goal of a sequence to the next fluctuated between very high and very low levels. This fluctuation made it difficult to analyze the *trend* of retrievals. We gained a better perspective by graphing "successive averages" of the percentage considered: the x-axis represents the course of the run as before, but each value on the y-axis is the average of the percentages from the beginning of the sequence to that point. Figure 5 shows typical results: twenty runs with introspective re-indexing on the left and twenty without it on the right. The averages with re-indexing declined over time to a much lower level than those without: 30-45% instead of 60-70%. A similar pattern of decline appeared in every sequence except the anomalous one. In some cases the decrease was much more dramatic, both in how low the percentage became and in how consistently runs displayed that low percentage. The best performance correlated with sequences with higher overall success rates, and with those having a greater *difference* between the success rates with and without re-indexing.

5.3 The Anomalous Sequence

The most compelling explanation for the anomalous sequence is that the introspective learning simply did not create new features which would be useful in later situations. This is supported by the anomalous sequence's retrieval efficiency with and without re-indexing: the average percentage considered in retrieval did not change significantly with or without introspective re-indexing. This suggests that, for this case, the new features learned introspectively did not apply to later situations and so could not provide any benefit. Consequently, the features learned through introspective reasoning did not group together useful cases, but rather caused an occasional additional failure by restricting retrieval so that a useful case might not be considered at all. This limiting effect of re-indexing would be magnified by the small size of the initial case base, together with the fact that few cases were being learned during the course of the run. Because ROBBIE's re-indexing depends on having a case in memory which shares important features with the current solution, the fewer the cases in memory, the less likely a good match will be found and a useful feature learned. Characterizing sequences like this one further and determining their frequency is an important task for the future.

5.4 Summary of Benefits and Costs

From the results discussed above, we can conclude that introspective reasoning does provide a tangible benefit under most circumstances, though further study is needed to characterize its failure for one sequence. In the other trials, the benefits we expected were demonstrated across a broad set of examples. ROBBIE generated more successful plans when using re-indexing, while at the same time improving the efficiency of its retrieval. While the percentage of cases considered in retrieval with re-indexing remained at a high level for goals which were dissimilar to cases in memory, the average percentage declined to a significantly lower level than with case learning alone.

We must address the other side of the coin: the costs of introspective reasoning. Monitoring for failures and explaining them require some amount of processing overhead; if it were sufficiently high it might make applying introspective reasoning infeasible. Re-indexing also takes time to reconsider cases in memory. However, in ROBBIE we have seen no severe performance degradation because of the use of introspective reasoning. Introspective reasoning can improve the efficiency of the system's reasoning, in this case by reducing the effort of considering similar cases. If ROBBIE's plans were executed in the real world, the execution time lost by selecting the wrong plan and recovering could overwhelm the cost of introspective reasoning which would avoid repeating such failures. Combining these advantages with the extension of the set of problems the system can solve, introspective reasoning seems highly promising.[4]

6 Relationship to previous research

There has been a significant amount of work in the past few years on methods for implementing introspective reasoning. Our work was inspired by the proposals for using model-based reasoning to improve the performance of CBR systems made by [Birnbaum *et al.*, 1991]. In a similar spirit, Freed implements an introspective model in RAPTER, a reactive planning system[Freed and Collins, 1994]. RAPTER focuses on analysis of the reasoning trace of the planner, using "justification structures" which trace the chain of assumptions made by the system used to determine what to repair.

Stroulia uses a "Structure-Behavior-Function" model, a form originally designed for diagnosis of device failures,

[4]The cost in development time expended to construct the introspective model is another issue, see [Fox and Leake, 1995] for a detailed discussion of model transfer.

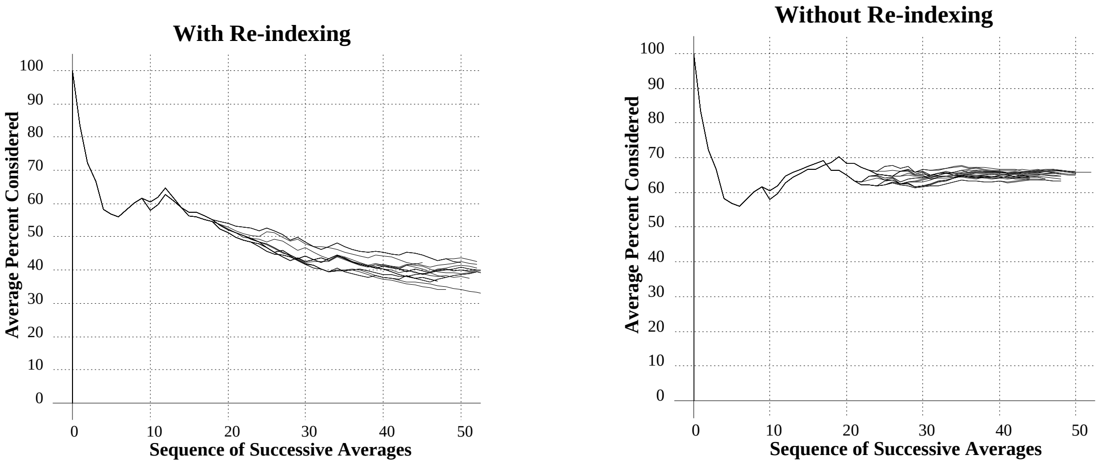

Figure 5: Successive averages show that without re-indexing, retrieval considers a stable high percentage, but with re-indexing, performance drops to a lower average percentage, with greater variation. (50% permuted sequence)

to perform introspective reasoning about an underlying system[Stroulia and Goel, 1994]. Her introspective reasoner, Autognostic, has been applied to two underlying systems, showing that transfer of introspective models is feasible. The SBF model provides strategies for assigning blame for the failure, and provides information about the device being modeled beyond that included in ROBBIE's model. ROBBIE's model is designed to support failure detection as well as recovery, and facilitates that process through the types of assertions included and their organization.

Oehlmann integrates introspective reasoning with the overall domain task in IULIAN: discovering answers to questions through experience or experimentation [Oehlmann *et al.*, 1994]. IULIAN contains plans for solving domain tasks, and introspective plans for controlling its reasoning process. It learns by applying these plans to its domain and reasoning tasks, and remembering the resulting successes or failures.

Cox implements introspective reasoning in Meta-AQUA by maintaining a set of reasoning trace templates (Meta-XPs) that describe different reasoning failures, and detecting reasoning failures that match a template[Ram and Cox, 1994]. Meta-XPs provide the means for determining the failure and suggestions for repairs.

In CBR research outside of introspective reasoning, multiple methods have been proposed for determining relevant indices. For example, explanations are often used to determine the relevance of features when assigning indices to new cases (e.g., [Barletta and Mark, 1988; Leake and Owens, 1986; Ram, 1993]). However, less attention has been devoted to the central questions addressed by ROBBIE: when and how an existing case in memory should be re-indexed. Some approaches alter indices in response to external feedback [Redmond, 1992; Veloso and Carbonell, 1993]; ROBBIE instead uses after-the-fact introspective analysis of its reasoning performance to determine whether new indices should be learned. In addition, when ROBBIE chooses new indices, it uses knowledge of both the faulty retrieval and the case that should have been retrieved to guide the choice of new indices. This allows it to select indices that are useful to discriminate between the cases currently in its memory [Fox and Leake, 1994].

7 Conclusions

The ROBBIE system combines introspective reasoning with a case-based planner to create a system that not only learns new plans in response to its experiences, but also learns to plan better by learning new features to use in indexing. ROBBIE uses a declarative model of the case-based reasoning process which provides expectations about its reasoning performance through assertions about the ideal reasoning and results. This model contains both general and specific assertions; the framework is designed to be applicable to other underlying systems and failures [Fox and Leake, 1995].

In previous research, little has been done to quantitatively evaluate introspective reasoning beyond isolated examples. We have performed empirical analysis of ROBBIE to determine its performance by measuring the success rate and the reasoning efficiency. We discovered that across a range of inputs, ROBBIE with introspective re-indexing improved the efficiency of its retrieval process by learning features to constrain the cases considered and to focus on the best cases. With one exception, ROBBIE was also more successful in creating plans when using introspective re-indexing than when doing case learning alone.

Many questions still remain to be answered. Perhaps most important is a detailed analysis of the reasons for ROBBIE's successes, and its failure. We must characterize the sequence which caused ROBBIE to perform worse with introspective re-indexing, and determine how common such a failure might be. We believe that the failure is due to the sparse introspective learning opportunities with a small case base that remained small due to many failures over time, combined with the possible over-constraining of the retrieval process due to learning irrelevant features.

As future research, we plan to expand the model of ROBBIE to include diagnosis and repair of other portions of the CBR system, such as the weighting of features in retrieval, the choice of strategies for adaptation, and the choice of plan execution steps. We plan eventually to transfer the introspective framework we have developed to other underlying systems, and to consider more qualitative measures of ROBBIE's introspective reasoner such as the quality of reasoning, the quality of plan produced, and its value as a cognitive model.

While much work remains to be done in creating the best method for introspective reasoning and evaluating its performance, the work described here clearly demonstrates that learning based on introspective reasoning is a feasible addition to a case-based planning system such as ROBBIE and provides a tangible benefit over fixed reasoning. We are encouraged by ROBBIE's success to expect introspective learning to provide more advantages for CBR and elsewhere in the future.

References

[Barletta and Mark, 1988] R. Barletta and W. Mark. Explanation-based indexing of cases. In J. Kolodner, editor, *Proceedings of a Workshop on Case-Based Reasoning*, pages 50–60, Palo Alto, 1988. DARPA, Morgan Kaufmann, Inc.

[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 Case-Based Reasoning Workshop*, pages 215–224, San Mateo, 1991. DARPA, Morgan Kaufmann, Inc.

[Bradzil and Konolige, 1990] P. Bradzil and K. Konolige, editors. *Machine Learning, Meta-Reasoning and Logics.* Kluwer Academic Publishers, Boston, 1990.

[Collins *et al.*, 1993] G. Collins, L. Birnbaum, B. Krulwich, and M. Freed. The role of self-models in learning to plan. In *Foundations of Knowledge Aquisition: Machine Learning*, pages 83–116. Kluwer Academic Publishers, 1993.

[Cox and Freed, 1995] M. Cox and M. Freed. Using knowledge of cognitive behavior to learn from failure. In *Proceedings of the 1995 AAAI Spring Symposium on Representing Mental States and Mechanisms*, 1995. (in press).

[Davis, 1982] R. Davis. Application of meta level knowledge to the construction maintenance and use of large knowledge bases. In R. Davis and D. Lenat, editors, *Knowledge-Based Systems in Artificial Intelligence*, pages 227–484. McGraw-Hill, 1982.

[Firby, 1989] R. J. Firby. *Adaptive Execution in Complex Dynamic Worlds.* PhD thesis, Yale University, Computer Science Department, 1989. Technical Report 672.

[Fox and Leake, 1994] S. Fox and D. Leake. Using introspective reasoning to guide index refinement in case-based reasoning. In *Proceedings of the Sixteenth Annual Conference of the Cognitive Science Society*, pages 324–329, Atlanta, GA, 1994. Lawrence Erlbaum Associates.

[Fox and Leake, 1995] S. Fox and D. Leake. Modeling case-based planning for repairing reasoning failures. In *Proceedings of the 1995 AAAI Spring Symposium on Representing Mental States and Mechanisms*, Stanford, CA, March 1995. AAAI. (ftp.cs.indiana.edu:/pub/leake/p-95-02.ps.Z).

[Freed and Collins, 1994] M. Freed and G. Collins. Adapting routines to improve task coordination. In *Proceedings of the 1994 Conference on AI Planning Systems*, pages 255–259, 1994.

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

[Leake and Owens, 1986] D. Leake and C. Owens. Organizing memory for explanation. In *Proceedings of the Eighth Annual Conference of the Cognitive Science Society*, pages 710–715, Amherst, MA, August 1986. Cognitive Science Society.

[Oehlmann *et al.*, 1994] R. Oehlmann, P. Edwards, and D. Sleeman. Changing the viewpoint: Re-indexing by introspective questioning. In *Proceedings of the Sixteenth Annual Conference of the Cognitive Science Society*, pages 675–680. Lawrence Erlbaum Associates, 1994.

[Ram and Cox, 1994] A. Ram and M. Cox. Introspective reasoning using meta-explanations for multistrategy learning. In R. Michalski and G. Tecuci, editors, *Machine Learning: A multistrategy approach Vol. IV*, pages 349–377. Morgan Kaufmann, 1994.

[Ram, 1993] A. Ram. Indexing, elaboration and refinement: Incremental learning of explanatory cases. *Machine Learning*, 10(3):201–248, 1993.

[Redmond, 1992] M. Redmond. *Learning by Observing and Understanding Expert Problem Solving.* PhD thesis, College of Computing, Georgia Institute of Technology, 1992. Technical report GIT-CC-92/43.

[Riesbeck and Schank, 1989] C. Riesbeck and R.C. Schank. *Inside Case-Based Reasoning.* Lawrence Erlbaum Associates, Hillsdale, NJ, 1989.

[Stroulia and Goel, 1994] E. Stroulia and A. Goel. Task structures: What to learn? In M. desJardins and A. Ram, editors, *Proceedings of the 1994 AAAI Spring Symposium on Goal-driven Learning*, pages 112–121. AAAI Press, 1994.

[Veloso and Carbonell, 1993] M. Veloso and J. Carbonell. Derivational analogy in prodigy: Automating case acquisition, storage, and utilization. *Machine Learning*, 10(3):249–278, 1993.