JOURNAL OF INTELLIGENT SYSTEMS

AN INTERDISCIPLINARY JOURNAL OF ARTIFICIAL INTELLIGENCE, COGNITIVE SCIENCE AND NEUROSCIENCE

AIMS AND SCOPE

The journal's aim is to provide research and review papers on an interdisciplinary level, where the focal point is the field of intelligent systems. The field includes: artificial intelligence techniques and theories, models and computational theories of human cognition, perception and motivation; brain models, artificial neural nets and neural computing. It covers the contributions of the social and human sciences, as well as computer science, to the analysis and application of information technology. The field also covers theoretical analysis of possible systems which could display intelligence, questioning how technological systems advance our understanding of intelligent systems and addressing philosophical questions that arise.

Scientific progress and an interdisciplinary understanding is needed as the basis of new technology. However, those sciences which are concerned with the understanding of human intelligence stand to gain by a dialogue with those developing the new technology. For in developing a complex technological system it is often necessary to place theoretical concepts into explicit formulations that make them more clear.

Three main types of papers are published: **Research papers**, which provide primary detailed information on original research; **reviews**, which contribute to the integration of existing knowledge; and **brief communications**, including short essays and book reviews.

Contributions of CBR to an Integrated Reasoning System

JAMES M. SKINNER and GEORGE F. LUGER

AI Technology Office, Wright Laboratory, Dayton, OH 45433, USA

and

Department of Computer Science, University of New Mexico,
Albuquerque, NM 87131, USA

CONTENTS

SYNOPSIS

Our research (Skinner 1992, Skinner & Luger 1992) demonstrates that combining multiple reasoning methodologies in a hybrid system allows the strengths of one methodology to compensate for the weaknesses of others. In addition, we have documented four aspects of interaction between the methodologies that can lead to synergistic behavior: *cooperation, confirmation, refutation,* and *follow-up*. Of the four reasoning methodologies selected for our research (conventional, rule-based, model-based, and case-based), case-based reasoning was remarkable in that it was the only form of reasoning that participated in all four aspects of interaction. This paper describes our experiment in integrating reasoning paradigms and documents the utility of the case-based reasoner through the four aspects of synergistic behavior observed.

1. INTRODUCTION

Earlier research (Skinner 1988, Skinner & Luger 1991) strongly suggested that the best approach to many problem-solving tasks may not be through a single method of reasoning, but rather through the blending of several diverse reasoning methodologies. In (Skinner & Luger 1992) we describe in detail an architecture that integrates diverse reasoning methodologies in a manner that allows the system to benefit from the strengths of each, while minimizing their respective weaknesses. We selected four complementary reasoning methodologies for integration based on their potential utility in problem solving: conventional, rule-based, model-based, and case-based reasoning.

Conventional reasoning (CR) refers to procedural or algorithmic methods of problem-solving. While conventional systems are not normally associated with reasoning, they can perform in an intelligent manner by integrating reasoning with search techniques. *Rule-based reasoning* (RBR) applies empirical associations as a set of IF-THEN statements to known facts to infer new facts. Rule-based reasoning can be used to represent the experiential knowledge of a domain. *Model-based reasoning* (MBR) is based on representations of the physical structure of the domain. This technique employs an explicit model of the domain that depicts knowledge of the structure, function, and behavior of a system. MBR uses causal reasoning, reasoning from first principles, and reasoning from the principle of locality to solve problems about the domain. *Case-based reasoning* (CBR) uses past problem-solving episodes to solve new cases. It was this form of historical knowledge that we found to be predominant in the synergistic aspects of our integrated reasoning system, and is the topic of this paper.

We evaluated each of these four reasoning methodologies to determine their individual strengths and weaknesses. Next, we reviewed the literature for past approaches to integrating reasoning methodologies to identify improvements that can arise from supporting one reasoning approach with a secondary approach. We used these results along with established principles for controlling blackboards to design an architecture that supports the integration of diverse methodologies. We then constructed a proof-of-concept prototype for diagnosing faults in a subsystem of the Hubbell Space Telescope. During testing, we observed four aspects of a synergistic effect produced by this integrated approach to problem solving: *cooperation, confirmation, refutation*, and *follow-up* (these terms are described in Section 3.2). CBR was remarkable in that it was the only form of reasoning that participated in all four aspects.

The remainder of this paper highlights our experience with CBR throughout the project. Section 2 discusses CBR and past approaches to its combination with other methodologies. Section 3 presents an alternative problem-solving approach based on the blackboard architecture. Section 4 describes the prototype we developed to test this approach and Section 5 provides a sample operation of the prototype. Section 6 analyzes the performance of CBR in the sample operation and Section 7 provides our conclusions on employing CBR in an integrated environment.

2. CASE-BASED REASONING

Case-based reasoning (CBR) is the process of using the results of past problem-solving episodes (cases) to analyze or solve a new problem (Rissland & Skalak 1991). By relying on past cases, the quality and efficiency of the reasoning is increased through the derivation of shortcuts and the anticipation of problems in new situations. Newly solved cases can be added to memory allowing the system to continually improve performance through "learning".

CBR evolved from Roger Schank's research in the mid-seventies on natural language processing in which he invented the concept of a "script". A script is a structured representation describing a stereotyped sequence of events in a particular context. Scripts are used to resolve ambiguities in language by filling in missing information and providing default assumptions. When Schank extended his work to planning, he theorized that people do not construct plans from first principles. Rather, they find the best plan they know of and attempt to adapt that plan to the current situation. This became the foundation for CBR (Reisbeck & Schank 1989).

CBR systems comprise a case memory, indices into the memory, metrics for assessing the similarity or relevancy of a case, and procedures for adapting past cases to new situations. Each case in the case base is a set of empirical circumstances representing a past problem-solving session. The case base not only stores successes for reuse, but also failures to avoid repeating mistakes. Cases in memory are indexed via salient features. Determination of the salient features, as well as metrics for assessing similarity is domain dependent.

Descriptions of the operation of CBR systems differ, but most are described as having five steps (Rissland & Skalak 1991): (1) recall relevant cases from case memory, (2) select the most promising case(s) to reason with, (3) construct a solution for the new case, (4) test and criticize the proposed solution, and (5) update the case memory and adjust the indexing mechanisms.

The two main types of CBR are precedent-based and problem-solving. Precedent-based CBR, similar to legal reasoning, argues that a new situation should or should not be treated like a past case based on similarities or differences between the cases. Problem-solving CBR, used for tasks such as design or planning, formulates a solution suited to the new case by modification of past solutions.

Proponents of CBR contend that it is only natural to use cases for automated reasoning because "case-based reasoning is the essence of how human reasoning works" (Reisbeck & Schank 1989). These proponents assert that human experts reason with cases to explain, persuade, learn, teach, and plan (Ashley & Sycara 1991). This tendency of experts to rely on cases simplifies knowledge acquisition and provides a starting point or "seed" for many domains. Another advantage is that no causal model of the domain is necessary. CBR becomes a sensible method of compiling past solutions to avoid "reinventing the wheel" or repeating past mistakes. The resulting shortcuts in reasoning and the capability of avoiding past errors enhance performance.

The difficulties of employing CBR include the complexity of handling a large number of cases, the implementation details of CBR theory, and a lack of robustness due to the absence of fundamental knowledge of the domain. Because CBR systems tend to require a large number of cases, the cases must be organized and indexed to minimize search and implement efficient retrieval. Implementation requires algorithms for selecting the 'best' match, modifying past cases, anticipating problems, repairing faulty cases, and testing the results.

Most successes in CBR have been in domains involving classification such as the medical and legal domains, or problem-solving tasks such as design, planning, and diagnosis (Ashley & Rissland 1987). CBR is best suited to applications in which many training cases are available and where it may be difficult to specify appropriate behavior using abstract rules. A list of CBR's strengths and weaknesses is incorporated in Table 1. This table summarizes the results of our evaluation of the characteristics of the four reasoning methodologies and is documented in Skinner (1992).

2.1 Past Approaches of Employing CBR in an Integrated Reasoning System

A review of the literature revealed several instances in which researchers found that their CBR systems could supplement, or be supplemented by, other reasoning methodologies. A summary of CBR's contributions to RBR and MBR is incorporated in Table 2. This table reflects the lessons learned from the survey of past approaches to integrating reasoning paradigms and is documented in Skinner (1992).

Rules can be used in many aspects of the CBR process. For example, CHEF, a planning system for Chinese recipes, uses rules in at least three of its five functional components: the MODIFIER, the ASSIGNER, and the ANTICIPATOR. The MODIFIER has a library of modification rules that can add steps to a plan to achieve a goal. The ASSIGNER creates a set of rules that mark features in a situation as predictive of problems that arose in that situation. The ANTICIPATOR uses this set of rules in the early stages of planning for the anticipation of problems (Hammond 1989).

Although not as common in practice, cases can support rules to: (1) annotate rules with exceptions, counter examples, and exemplars; (2) explain or justify rules with case examples; (3) draw inferences from cases where rules are ill-defined; (4) test and evaluate rules; and (5) record rule-based solutions for future short cuts (Rissland & Skalak 1991). For example, ANAPRON, a system for pronouncing surnames, uses cases to reason about exceptions to the rules in a RBR system (Golding & Rosenbloom 1989). While using rules as a default, ANAPRON checks aspects of the problem against known exceptions, and if compelling

TABLE 1
Characteristics of Reasoning Methodologies (Summary)

Method	Advantages	Disadvantages
CBR	ability to employ historical knowledge allows shortcuts in reasoning avoids past errors no domain model required existing cases for some domains knowledge acquisition relatively easy coding relatively easy clever indexing can add insight	lacks fundamental knowledge of domain complexity issues with large case base hard to define criteria for matching hard to define criteria for indexing difficult to construct/maintain index
RBR	ability to employ experiential knowledge modularity eases construction & maintenance high performance possible in limited domain simple method of providing explanations rules map naturally onto search space rules are easier to trace and debug steps in process are open to inspection separation of knowledge/control	lacks fundamental knowledge of domain cannot solve unforeseen problems rapidly degrades near edges of domain explanations often inadequate knowledge is task dependent difficult to verify heuristics multiple experts may disagree handling exceptions to rules
CR	ability to employ procedural knowledge correct answers when problem is constrained proven V&V techniques exist simple implementation	must have algorithm for task difficult to incorporate heuristics
MBR	ability to employ structural knowledge robust knowledge transferable between tasks can provide causal explanations versatile	lacks experiential knowledge of domain CPU/ time intensive requires an explicit domain model

TABLE 2
Improvements to Primary Reasoning Methodology by Supporting Methodology

Supporting Method	Primary Method: Cases	Primary Method: Rules	Primary Method: Models
Cases	N/A	provide exceptions provide interpretation provide examples assist incomplete rules assist inconsistent rules provide counterexamples provide exemplars provide explanations provide justification test/evaluate rules record for future use provide backup to failure	record explanations record solutions provide backup to failure
Rules	bootstrap CBR process anticipate problems improve indexing modify cases evaluate solution verify solution backup to failure	N/A	reduce response time heuristic search of model reduce CPU use focus reasoning add experiential knowledge provide backup to failure
Models	improve explanations backup to failure	enhance explanations improve robustness provide backup to failure	N/A

similarities are noted, the problem is modelled after the exception rather than the rule.

In another example, the CAse-BAsed REasoning Tool (CABARET) employs rules and cases to reason about circumstances under which a taxpayer may legitimately deduct expenses relating to an office maintained at home. CABARET's case knowledge base consists of 29 actual and hypothetical cases indexed on 14 dimensions. The rule base contains 10 home office deduction rules representing the governing law and related condition-action rules (Rissland & Skalak 1991).

Finally, systems integrating CBR and MBR are much less common than those combining CBR and RBR. CASEY is an example of a system that integrates CBR and MBR to manage patients with cardiac diseases (Koton 1988). CASEY uses CBR to store and retrieve causal explanations of the findings from a model-based expert system known as the Heart-Failure Program in significantly less time than if generated directly from the model.

3. AN ARCHITECTURE FOR INTEGRATING REASONING PARADIGMS

We used the results from the study of the individual reasoning methodologies and the survey of past integration approaches to modify the traditional blackboard architecture so that it allows diverse reasoning methodologies to be integrated. This section reviews the traditional blackboard architecture, the fundamental modification necessary to allow the integration of reasoning paradigms, and the control algorithm required for implementation.

3.1 The Traditional Blackboard Architecture

The blackboard model shown in Figure 1 consists of three components: the blackboard data structure, one or more knowledge sources,

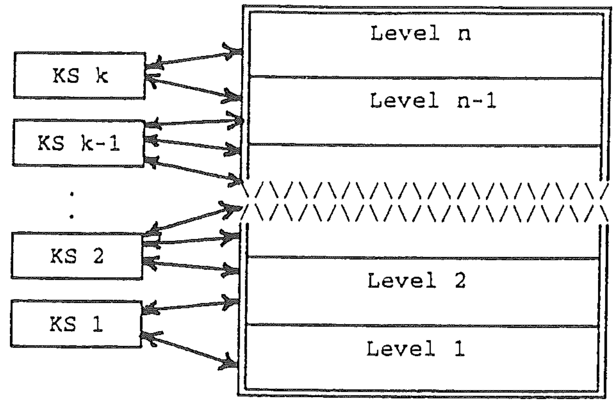

Fig. 1: Traditional blackboard architecture.

and the control component. The blackboard data structure is a global data store that holds the problem-solving state data. The blackboard can be partitioned into distinct information levels to allow information about the problem to be represented at different levels of detail. For example, the first blackboard, Hearsay-II, used the levels to represent phrases, words, syllables, segments, and other building blocks of the spoken language. All communication and interaction of the system takes place solely through the blackboard (Engelmore 1988).

The domain knowledge needed to solve the problem is partitioned into independent knowledge sources. Knowledge sources produce changes to the blackboard by transforming information on one level to information on the same or other levels. Typically, the knowledge is stored as condition-action pairs.

Control can reside in the knowledge sources, on the blackboard, in a separate module, or in some combination of these approaches. Although implementations vary, operation of the blackboard usually begins with a change to the blackboard caused by the posting of the initial problem statement. Each change to the blackboard constitutes an event that in the presence of other information on the blackboard can trigger one or more knowledge sources. The control mechanism selects a single knowledge source to execute its action on each problem-solving cycle. When a knowledge source is triggered, it will typically produce new blackboard events. These events may in turn trigger other knowledge sources (Hayes-Roth 1988).

A common analogy used to describe the behavior of the blackboard model is that of a group of people trying to assemble a jigsaw puzzle (Engelmore 1988). Imagine a room with a large sticky blackboard and a group of people each holding pieces of a jigsaw puzzle. Volunteers first put their most promising pieces on the board. Each member of the group looks at his or her pieces to see whether any fit with the pieces already on the board. If so, those with appropriate pieces update the evolving solution. The new updates cause other pieces to fall into place, and other people add their pieces. The whole puzzle can be solved in complete silence – there is no need for direct communication between the individuals. The apparent cooperative behavior is mediated by the state of the solution on the blackboard. The solution is built incrementally and opportunistically.

3.2 The Synergistic Reasoning System

We have designed an alternative problem-solving approach based on the blackboard problem-solving model which we call the Synergistic Reasoning System (SRS). The difference between SRS and the traditional blackboard model can be understood by contrasting analogies. An analogy for SRS is a person taking a closed-book test. All of the knowledge to be used during the test is self contained. However, the person is likely to use several different methods of reasoning while taking the test including relying on past experiences, employing heuristics, following procedures, or developing a mental model of a problem.

Implementing this approach requires a fundamental modification of the blackboard model. Rather than partitioning the domain knowledge functionally, we partition the problem-solving approach in terms of reasoning methodologies. Therefore, instead of the traditional knowledge sources, the system has reasoning modules. Specifically, the synergistic reasoning system is equipped with four reasoning modules: case-based, rule-based, conventional, and model-based. The system dynamically switches between the reasoning modules as necessary to solve the problem.

This approach produces a synergistic effect through *cooperation, confirmation, refutation,* and *follow-up. Cooperation* allows the individual reasoning modules to post partial solutions, enabling the system to solve problems that could not be solved by any single module. Thus, one module may post a partial solution not obtainable by any of the other modules, and while this module might not be able to generate the entire solution, one of the remaining modules may be able to generate the desired solution based on this partial result.

Confirmation allows reasoning modules to verify results from other modules. As an example, when the RBR recommends a tactic for solving a problem, the CBR may be able to provide past cases in which the tactic was successful. Confirmation can be used to increase the confidence of the user or to choose between competing tactics that have been proposed for resolving a problem under consideration.

Refutation is the ability of one reasoning module to refute conclusions of another module. That is, while incomplete information may cause one reasoning module to arrive at an incorrect conclusion, a second module may have information that disputes this conclusion. Using the same example as above, the CBR may be able to demonstrate that past attempts at solving the problem with the tactics proposed were unsuccessful. Again, this would be useful to increase confidence or to select between competing proposals.

Follow-up was the most interesting of the aspects observed. Follow-up searches for trends in the conclusions of the system indicative of deeper problems; it then uses these trends to continue diagnosis. For example, the CBR can be used to detect repeated adjustments that temporarily relieve a problem. This repeated occurrence of the problem is then seen as symptomatic of a deeper problem and is added as a new symptom for further diagnosis.

The concept of follow-up is best illustrated through a concrete example. Consider the testing of inertial navigation systems (INS). Suppose that the automatic test equipment for a system reports that the gyroscope is the source of the problem in an INS unit. The technician's confidence in this information is likely to be high, because gyroscopes are the most common failure in such systems. If direct tests on the gyroscope reveal that it has a bad output, the technician's confidence is likely to be increased. If after the faulty gyroscope has been replaced the unit performs properly, the unit will probably be fielded with a high level of confidence that the problem has been isolated and corrected.

But what if we provide the technician with the historical knowledge that this same unit has been serviced six times in the last five months, and that each time the fault was determined to be the gyroscope. The technician will probably add this trend as a symptom and continue to diagnose the problem. It may be found that the system has a deeper problem, perhaps that a voltage regulator is faulty, causing the gyroscopes to fail. This identification of trends and the subsequent search for an underlying cause is the objective of follow-up.

3.3 Controlling Multiple Reasoning Paradigms

We derived a control algorithm suitable for a synergistic approach by combining principles for controlling blackboards from Hayes-Roth & Lesser (1977) with the findings from the survey of the reasoning methodologies. This algorithm is exercised by an Executive Module (EM) which is responsible for coordinating the problem-solving process. Details of the control algorithm can be found in Skinner (1992); a somewhat more intuitive description is presented here.

In blackboard terms, scheduling knowledge sources to minimize the number of steps in a problem-solving session is known as the *focus of attention*. The developers of HEARSAY-II identified five fundamental principles for controlling the focus of attention (Hayes-Roth & Lesser 1977). Although these principles are defined in terms of knowledge sources, we have adapted them to the control of reasoning modules. The principles are:

(1) *The competition principle*: the best of several local alternatives should be performed first. This governs behavioral options which are locally competitive in the sense that a definite outcome of one may obviate the others.

(2) *The validity principle*: knowledge sources operating on the most valid data should be executed first. If everything else is equal, the preferred knowledge source should be the one working with the most credible data.

(3) *The significance principle*: knowledge sources whose responses are most important should be executed first. This principle ensures the most important steps are performed first.

(4) *The efficiency principle*: knowledge sources which perform most reliably and inexpensively should be executed first.

(5) *The goal satisfaction principle*: knowledge sources whose responses are most likely to satisfy processing goals should be executed first.

When applied in the context of SRS, these principles led to the following set of general heuristics. From principle (1), recommendations should be followed in order of their specificity, likelihood, and frequency. From principle (2), recommendations should be executed according to their confidence values. From principle (3), suspected catastrophic or time critical recommendations should be capable of preempting other tasks. From principle (4), the most efficient reasoning modules should be used first. From principle (5), top-level goals should have priority over sub-goals.

Additional heuristics derived from the advantages of each methodology shown in Table 1 suggest that: (1) CR should be used whenever a polynomial-time algorithm exists; (2) CBR should be used if no specific recommendation is present and as a means of error checking; (3) each of the reasoners should be used as a failure backup to the others; (4) RBR should be used for quick fixes, if no causal explanation is required, or if time constraints are strict; (5) MBR should be used if a causal explanation is required, but only if adequate time is available; and (6) the results of the sessions should be stored by the CBR module.

The result is a set of 18 guidelines tailored for a diagnostic application. These guidelines are by no means static – the intent is for the set to grow and to be refined as dictated by results of continuing research in integrating reasoning paradigms. Examples of the guidelines are shown below:

(a) If more than one reasoner can act on a goal,
employ in the order of RBR, CBR, MBR.
(b) If no specific recommendation is present,
employ in the order of CBR, MBR, RBR.
(c) If the diagnosis session is complete,
employ CBR to store session results.
(d) If a fault has been diagnosed,
initiate confirmation, refutation, and follow-up.

Guideline (a) presents the criteria for choosing between competing reasoning modules to perform a task. The priority used is to rely on RBR, then CBR, then MBR. In general, robustness increases and efficiency decreases in order of CBR, RBR, and MBR. By favoring RBR, a balance between robustness and efficiency is achieved. CBR is selected second because it can be executed quickly.

Guideline (b) states that when no specific recommendations are present, the reasoning modules are prioritized as CBR first, then MBR, then RBR. The rationale for the prioritization is that the information available favors CBR over MBR, and MBR over RBR. The case base is indexed by symptoms and at least one symptom is guaranteed to be present; otherwise the CR module would be in control. If the CBR is unsuccessful, the list of suspected components provides a focus for the MBR to diagnose the fault. If still unsuccessful, a goal is created for the RBR to diagnose the current list of symptoms.

Guideline (c) dictates the problem-solving information be stored for use in future diagnostic sessions. This is accomplished by the CBR module and provides a simple means of machine learning. Guideline (d) is a result of this research and the discovery of how SRS can produce synergistic behavior. It is implemented by posting the three goals when a diagnosis is reached. Each module then responds according to its ability.

4. THE SRS PROTOTYPE

We constructed a prototype of the Synergistic Reasoning System (SRS) for diagnosing faults in the Hubbell Space Telescope (HST) Reaction Wheel Assembly (RWA). The function of the HST RWA is to point the Space Telescope at the proper area of the sky and keep the telescope locked onto its target. The RWA functions according to the principle of conservation of angular momentum. When the telescope is stationary, the reaction wheel moves at a small speed to counteract the torque caused by Earth's gravitational field. To move the telescope, the speed of the reaction wheel is increased, causing the telescope to spin in the opposite direction. When the telescope nears its proper orientation, the spin is reversed and the telescope slows down. There are four reaction wheels aboard HST, and the

and Temperature of Tunnel-Sensor is OK
THEN Set Malfunction of RCE-Bearing to True.

This rule states that if the sensor for RCE-bearing is abnormally high and nearby sensor readings are normal, then there must be a malfunction within the RCE-Bearing (Keller 1990).

The MBR module diagnoses the suspected components to determine the likely cause of the symptoms. The MBR module is implemented in CLOS and uses the principle of locality. This principle considers how components are connected either mechanically, electrically, or physically to determine how the behavior of one component can be influenced by another component (Davis 1985). The primary knowledge source for the models used in the prototype of the SRS was a set of papers written by researchers from NASA Ames and Stanford (Keller 1990, Gruber & Iwasaki 1990) that cover the structural and functional models for the HST RWA. Secondary sources used to provide details for constructing models included knowledge engineering sessions with a satellite operator (Garnham 1990) and a satellite analyst (Campbell 1992) to determine how failures in the system may reveal themselves as symptoms. One of the views of the domain model is shown in Figure 2.

The CBR module is implemented through GBB's pattern-matching facilities. Specifics about this module will be discussed in more detail in the next section. The Executive Module (EM) exercises explicit control over SRS by determining the order in which the reasoning modules work on the problem and coordinating the problem-solving process. It is implemented through a combination of GBB's control shell, knowledge sources, and CLOS.

4.2 The CBR Module and Cases for the Domain

An important factor in selecting the HST RWA as the domain for our research was a desire to use existing knowledge bases to minimize the sum of the torque forces generated by these wheels enables the telescope to rotate about an arbitrary axis (Keller 1990).

4.1 Structure for a SRS Prototype

The SRS prototype was implemented in Generic Blackboard (GBB), a commercial tool kit based on the Common LISP Object System (CLOS) (Blackboard Technology Group 1991). SRS is a modified blackboard architecture with a hierarchical blackboard database, an Executive Module, and four reasoning modules: CBR, RBR, CR, and MBR. The blackboard database has one root blackboard and four blackboards as interior nodes, one for each of the individual reasoning modules. The root blackboard has seven spaces: Status, Symptoms, Suspects, Actions, Diagnosis, Recommendations, and Goals. The interior blackboards each have a single space to record local information.

The CR module reasons about the environment as long as the status of the system is nominal. For the HST RWA, the knowledge required is the set of control algorithms that are currently used onboard the vehicle. For purposes of the prototype, the CR module implements a simulator for the attitude control system that enables the satellite to maintain its correct position and attitude. The simulator allows the user to change the attitude of the satellite relative to the Earth or to change the path of the satellite around the Earth. The simulator fires the thrusters as necessary to achieve the new position and reflects the changes through graphics on the screen.

The RBR module treats the symptoms and goals of the problem-solving session as facts, asserting them into its knowledge base. The rules are a set of diagnostic associations relating the readings of the temperature sensors to the possibility of faults in the bearings or electronics. An example of one such rule concerning the rotor control electronics (RCE) is:

IF Temperature of RCE-Bearing-Sensor is High,
and Temperature of RCE-Sensor is OK,

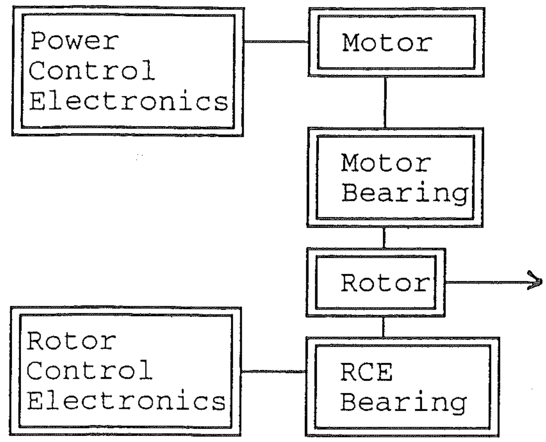

Fig. 2: One view of the domain model. This view reveals the mechanical connections of the Reaction Wheel Assembly.

temptation to build artificial synergism into the system. We were aware of existing rules and models for the domain, and because the HST had been fielded for over two years, we assumed that cases describing faults in the RWA could be obtained. When we contacted officials at NASA, we found that the performance of the HST RWA has been flawless. While this is excellent from an engineering standpoint, the result is that no cases existed, and had to be constructed based on consultations with a satellite analyst (Campbell 1992).

A sample case is shown in Figure 3. The cases are loaded by the CBR module when SRS is first initialized. They are stored on the CBR blackboard using indices representing the case number, the source of the case (either actual or hypothetical), the list of symptoms, the suspected components, the diagnosis, the list of actions taken to correct the fault, and the result (either successful or unsuccessful).

The CBR module participates in all four aspects of synergistic behavior. During cooperation, the CBR module is triggered by a Goal of

```
Case-Number: 2902248000 ;;; Dec 20 1991 1500
Symptoms: ((:weak-signal)
           (:calibrate-pointing :unsuccessful))
Suspect: none
Actions: ((:cr :symptom-posted :weak-signal)
          (:em :check-prior-messages :none)
          (:rbr :adjust-antenna :unsuccessful)
          (:cr :calibrate-pointing :unsuccessful)
          (:mbr :diagnose-acs :acs-faulty))
Diagnosis: attitude-control-system
Result: successful
Source: actual
```

Fig. 3: A sample case.

(:match-cases :symptom). The CBR module retrieves the set of cases with a successful diagnosis and in which one or more of the symptoms match a symptom in the current case. CBR then orders this set of possible-matches into groups from most symptoms in common to least symptoms in common. Beginning with the group that has the most symptoms in common, it determines the diagnosis common to the highest number of cases, then the diagnosis common to the second highest number of cases, continuing until all cases have been ordered. This is repeated for all groups, avoiding any repeated references to a component. The CBR module then returns a recommendation to diagnose the list of ranked components.

During confirmation the CBR module retrieves the set of successful-cases that have one or more of the same symptoms and the same diagnosis as the current case. The CBR records the Action as (:cbr :confirm :no-of-cases). During refutation the CBR module retrieves the set of failed-cases (those with Result = :unsuccessful) and from this set searches for past cases with the same set of symptoms as the current case and the same diagnosis. In this case, the Action will be posted as (:cbr :refute :no-of-cases).

During follow-up the CBR module retrieves the set of actual cases (those cases where Source = :actual) and matches on the Diagnosis slot of the past cases with the Diagnosis of the current session. If this problem has occurred recently, the multiple occurrence is added as a symptom and the follow-up is reported through the Actions as (:cbr :follow-up :continue-diagnosis).

The case number is the means by which we implement the concept of "recently". The case number is assigned by a function call to universal time. This call returns a single non-negative integer representing the number of seconds since midnight, January 1, 1900 GMT. The case number therefore serves as a unique identification number and a means by which the CBR module can employ temporal reasoning during follow-up. For the prototype, recent cases were defined as those which occurred in the past 24 hours. At the end of each problem-solving session, the results of the current session are stored by the CBR module.

5. SAMPLE OPERATION OF SRS

We developed a scenario to test the operation of SRS in which the onboard sensors detected a weak signal from the ground station. The response of the system is useful in depicting the four aspects of synergistic behavior. The state of the blackboard at various points is shown in the figures. An explanation of the events that led to these states follows.

The scenario begins during normal operations, with the CR module active. When the signal strength falls below a predetermined level, the CR module posts the symptom on the blackboard and surrenders control. The creation of a symptom causes the status to change to a fault condition which, in turn, triggers the EM.

After determining the symptom is not due to scheduled maintenance, the EM posts a goal to diagnose the symptom. The RBR responds using a set of rules for the antenna adjustment which allow it to increase the gain by ten percent or to calibrate pointing. The RBR recommends an increase in gain which boosts the signal and alleviates the problem. The diagnosis is low-gain and it would seem that the diagnostic session is complete. The posting of the diagnosis causes the EM to add three goals: confirm the diagnosis, refute the diagnosis, and follow-up on the diagnosis. This is SRS's method for error checking and increasing confidence in the conclusion. The state of the blackboard at this point is shown in Figure 4.

The CBR confirms the actions taken are the correct response for the given symptom by finding a past case which resulted in success. Next, the CBR attempts refutation, but cannot find any cases in which this tactic was unsuccessful. During follow-up, the CBR discovers that the gain has been increased twice in the last three hours. This trend, seen as indicative of a deeper problem, is posted as a new symptom and diagnosis is continued.

The RBR recommends calibrating the pointing of the antenna, but the CR module reports that calibration failed. This is added as a new symptom and causes SRS to suspect the Tracking, Telemetry, and Control (TT&C) subsystem. The state of the blackboard at this point is shown in Figure 5.

The MBR module constructs a model of the TT&C subsystem and checks each point, but no fault is found. At this point, the EM has no

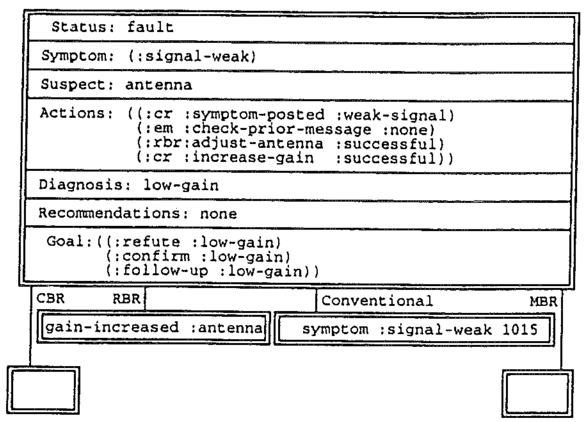

Fig. 4: Sample Operation. The upper portion of the figure shows the contents of the seven major spaces of the top-level blackboard. The lower four boxes reveal the contents of the individual modules.

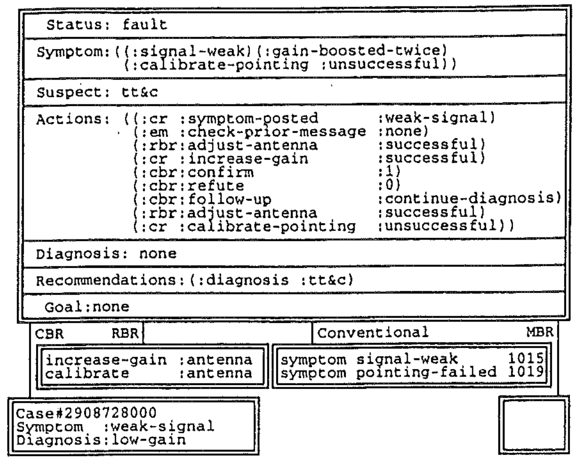

Fig. 5: Sample Operation (cont'd). Through *follow-up* and *cooperation* additional symptoms are identified. The CR module recommends that the TT&C subsystem be diagnosed.

specific recommendations and must rely on the predetermined guidelines. The CBR module is used to search for past cases, retrieving a case in which the antenna could not be calibrated due to a fault in the attitude control system (ACS).

The MBR builds a model of the ACS and isolates the fault to the RWA. It cannot, however, find any malfunction in the components of the RWA. The RBR module uses experiential knowledge to determine the faulty behavior is due to a high ambient temperature in the bay and recommends opening a louver to the outside to allow heat to dissipate. The diagnosis is posted as :closed-louver and, once again, confirmation, refutation, and follow-up are posted as goals.

No confirmation is found, but the CR refutes the diagnosis; according to its data the louver is open. The EM relies on the MBR to resolve the contradiction. In diagnosing a model of the thermal control system containing the louver, the MBR determines that the input to the louver motor is good, but the louver is closed. The louver's actuator motor is determined to be bad and a backup motor is employed. This final state of the blackboard is shown in Figure 6. The results are stored by the CBR module, the blackboard is scrubbed, and control is returned to the CR module.

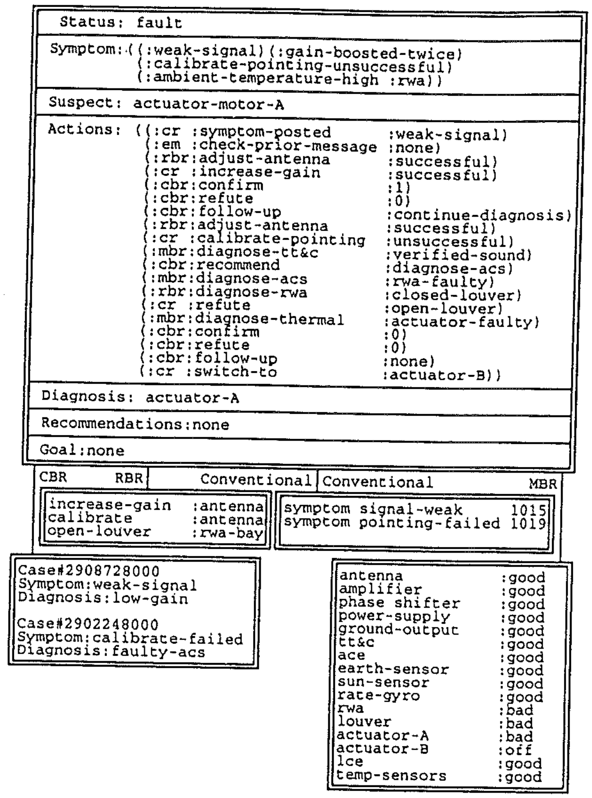

Fig. 6: Sample Operation Final State. The fault is isolated to the actuator motor. The results will be stored by the CBR module for use in future sessions.

6. CBR PERFORMANCE IN THE FOUR ASPECTS OF SYNERGISM

This example illustrates the four aspects of synergistic behavior:

cooperation, confirmation, refutation, and follow-up. Although the original symptom was a weak signal, the actual cause was the failure of an actuator motor in the thermal control system. This failure caused the louver to remain closed, thereby overheating the reaction wheel assembly. This in turn prevented the attitude control system from maintaining the correct attitude, causing the antenna to be improperly calibrated. As a result, the signal strength continually degraded, and a weak signal was observed.

Individually, none of the reasoners would have responded with a correct diagnosis. Both the CBR and RBR modules would have attributed a weak signal to low-gain. The CR module had no algorithm to solve the problem. The MBR module would have diagnosed the TT&C model, only to find all components were sound. Yet, by collaborating on the problem, the reasoning modules were able to produce a proper response.

The extent of CBR's role in the synergistic process was surprisingly high. Prior to constructing the prototype, it had been expected that the RBR or MBR module would be dominant, mainly because they held the bulk of the knowledge. While these systems were valuable, the CBR module was the one system that was involved in all four aspects of synergism.

Cooperation, the ability to construct a solution fro partial postings, was apparent as the reasoning modules worked together to isolate the problem. The CBR used historical knowledge to determine that the inability to calibrate the antenna could be due to a fault in the ACS. This allowed the reasoning process to continue even though no recommendations were present. Cooperation continued when the MBR used structural knowledge of the ACS to isolate the problem to the RWA, but, since heat flow was not included in the model, MBR was unable to determine the cause of the faulty behavior. The RBR used experiential knowledge to identify the source of the problem as a closed louver.

Confirmation, the ability of one reasoning module to verify the results of another module, was demonstrated when the CBR module increased the confidence of the decision to increase the gain of the antenna. Although this was only a temporary fix, it was the correct response for the available information.

The CBR module participated in refutation, but returned a negative reply. The case base did not contain any failed cases similar to the proposed solution. An example of successful refutation was demonstrated when the CR reported that the louver was already open. The RBR module had incomplete knowledge of the current configuration of the system, leading to an erroneous conclusion that the louver was closed. The additional information provided by the CR led to mediation of the contradiction by the MBR.

Finally, follow-up is the ability to identify trends indicative of deeper problems. This aspect of synergistic behavior occurred when the CBR discovered the repeated gain increase. Had this not been noted as a symptom of a deeper problem, an autonomous system might have continued to increase the gain, without addressing the underlying thermal problem which could eventually cause permanent damage.

7. CONCLUSIONS

Case-based reasoning can make an important contribution to an integrated reasoning environment; Table 2 lists some of the improvements that can be expected from adding cases to a rule-based or model-based system. Furthermore, CBR appears to be of particular importance in developing a system that exhibits synergistic behavior. In our research, CBR was the only form of reasoning that participated in all four of the aspects of synergism. The historical knowledge was found to be well-suited for use in implementing the concepts of cooperation, confirmation, refutation, and follow-up.

During cooperation, CBR was favored for first use and recovery. The efficiency of CBR made it an obvious candidate for initially addressing the problem. In addition, the CBR module was the most effective means of recovering from a state for which no recommendations were present. During confirmation and refutation, the historical form of knowledge employed by CBR was ideal for implementing an error-checking scheme that could result in the outright rejection of a conclusion, or simply an adjusting of its associated confidence value. During follow-up, the historical knowledge provided a means of identifying trends in the symptoms and conclusions of the reasoning system, and made it possible to add these trends as new symptoms to improve the diagnosis. Additionally, the time-stamp of the cases offered a means of temporal reasoning in the diagnosis. Finally, by storing problem-solving sessions in the case base, a simple but effective means of learning was implemented. This learning was critical for improving the performance during confirmation, refutation, and follow-up.

The current emphasis on CBR as well as hybrid systems ensures that research into employing CBR in a hybrid environment will continue. This research provides evidence that the result of integrating reasoning paradigms can be more than just the sum of the parts; it can result in synergistic behavior.

REFERENCES

Ashley K.D., Rissland E.L. Compare and Contrast, A Test of Expertise. *Proceedings of AAAI-87*: 273-278 (August 1987).

Ashley K.D., Sycara K.P. Case-based Reasoning Tutorial. *Ninth National Conference on Artificial Intelligence* (August 1991).

Blackboard Technology Group. GBB User's Guide Version 2.0. Amherst, MA: BBT Group, February 1991.

Campbell W.S., Satellite Analyst. Personal interviews. Kirtland AFB, NM, March 1992.

Davis R. Diagnosis Via Causal Reasoning: Paths of Interaction and the Locality Principle. In: J. Jeffrey Richardson, ed, *Artificial Intelligence in Maintenance*, Park Ridge, NJ: Noyes Publications, 1985; 102-122.

Engelmore R.S., Morgan A.J. *Blackboard Systems*. Wokingham, England: Addison-Wesley Publishing Company, Inc., 1988.

Garnham J. Satellite Operator. Personal interviews. Kirtland AFB, NM, October 1990.

Golding A.R., Rosenbloom P.S. Improving Rule-Based Systems through Case-Based Reasoning, *Proceedings: Case-Based Reasoning Workshop*, 1989; 22-27.

Gruber T., Iwasaki Y. How Things Work: Knowledge-based Modeling of Physical Devices, KSL 90-51. Stanford, CA: Stanford University, August 1990.

Hammond K.J. *Case-Based Planning, Perspectives in Artificial Intelligence Series*. San Diego, CA: Academic Press, Inc., 1989.

Hayes-Roth B., et al. BB1: An Implementation of the Blackboard Control Architecture. In: R. Engelmore, T. Morgan, eds, *Blackboard Systems*. Wokingham, England: Addison-Wesley Publishing Company, Inc., 1988; 231-244.

Iwasaki Y. Qualitative Physics. In: A. Barr, ed, *The Handbook of Artificial Intelligence*, Volume IV. Reading, MA: Addison-Wesley Publishing Company, Inc., 1989; 323-413.

Keller R., et al. Model Compilation: An Approach to Automated Model Derivation, NASA report RIA-90-04-06-1, April 1990.

Koton P. Reasoning about Evidence in Causal Explanations. *Proceedings of AAAI-88*, 1988; 256-261.

Luger G.F., Stubblefield W.A. *Artificial Intelligence: Structures and Strategies for Complex Problem Solving*. Redwood City, CA: Benjamin Cummings, 1993.

Perkins W.A., Austin A. Experiments with Temporal Reasoning Applied to Analysis of Telemetry Data, *Space Station Automation III*, Vol. 851. Society of Photo-Optical Instrumentation Engineers, 1987; 39-46.

Rissland E.L., Skalak D.B. CABARET: rule interpretation in a hybrid architecture. *International Journal of Man-Machine Studies*, 1991; 34: 839-887.

Reisbeck, C.K., Schank R.C. *Inside Case-based Reasoning*. Hillsdale, NJ: Lawrence Erlbaum Associates, 1989.

Skinner J.M. *A Diagnostic System Blending Deep and Shallow Reasoning*, MSCE Thesis, AFIT/GCE/ENG/88D-5, Air Force Institute of Technology, OH, December 1988.

Skinner J.M., Luger G.F. A Synergistic Approach to Reasoning for Autonomous Satellites. *Proceedings of the Advisory Group for Aerospace Research and Development (AGARD) Panel on Machine Intelligence for Aerospace Electronic Systems*, 1991.

Skinner J.M. *A Synergistic Approach to Reasoning*, Ph.D. Dissertation, Department of Computer Science, University of New Mexico, Albuquerque, NM 87131, May, 1992.

Skinner J.M., Luger G.F. An Architecture for Integrating Reasoning Paradigms. In: B. Nebel, C. Rich, W. Swartout, eds, *Principles of Knowledge Representation and Reasoning: Proceedings of the Third International Conference (KR92)*, San Mateo, CA: Morgan Kaufmann, 1992.

Wertz J.R., Larson W.J. *Space Mission Analysis and Design*. Dordrecht, Netherlands: Kluwer Academic Publishers, 1991.