Using Attention in Belief Revision

Xueming Huang, Gordon I. McCalla and Eric Neufeld

ARIES Laboratory
Department of Computational Science
University of Saskatchewan
Saskatoon, Saskatchewan, Canada S7N 0W0
huang@skorpio.usask.ca

Abstract

Belief revision for an intelligent system is usually computationally expensive. Here we tackle this problem by using *focus* in belief revision: that is, revision occurs only in a subset of beliefs under attention (or in focus). Attention can be shifted within the belief base, thus allowing use and revision of other subsets of beliefs. This attention-shifting belief revision architecture shows promise to allow efficient and natural revision of belief bases.

1. Introduction

Belief revision for an intelligent system is usually expensive. This is because it requires solving the intractable problem of detecting and removing contradictions that arise in the belief base when new beliefs are added in. The difficulty is amplified if minimal change of the belief base is required (Katsuno & Mendelzon 1989, Rao & Foo 1989). Thus, most belief revision systems require intractable computation (Martins & Shapiro 1988; Rao & Foo 1989). However, many applications require fast revision of their belief bases. In particular, if the application has a large belief base or if it is an on-line interactive system, such inefficient belief revision mechanisms would be intolerable.

We attack the problem of providing efficient, minimal revision of belief bases by using attention (or focus) in belief revision. The idea is that if in each belief revision session we only detect and remove contradictions in a small subset of beliefs under attention (in focus), then the required computation can be limited, even if it is exponential to the size of the subset under attention (consider 2^x, $x \leq C$, where C is a small constant). Attention can be shifted within the belief base, thus allowing use and revision of other subsets of beliefs. This idea is implemented in a system called the *attention-shifting belief revision system* (*ABRS*).

The paper is organized as follows: Section 2 presents a component of the ABRS, called the *evolutionary belief revision system* (*EBRS*), to attack the problem of minimal change in belief revision. Section 3 discusses the background and the trade offs involved in using attention in belief revision. Sections 4 and 5 describe the ABRS. Section 6 shows an example. Section 7 summarizes the discussions.

2. The EBRS

Two basic approaches are used in belief revision: *coherence belief revision* and *foundations belief revision* (Gardenfors 1990, Rao & Foo 1989). The fundamental difference between them is on the issue of whether justifications of beliefs should be taken into account during belief revision. The coherence approach focuses on minimal change to maintain logical consistency of the belief base, regardless of the justifications. The foundations approach insists that all beliefs must be well justified, namely, each belief must be either directly assumed by the system or supported by other justified beliefs, but the problem of minimal change is usually not addressed. Among existing belief revision systems, *propositional database updating systems* normally use the coherence approach (Dalal 1988, Weber 1986), while *reason maintenance systems* (*RMS's*) are considered foundations belief revision systems (Doyle 1979, Martins & Shapiro 1988).

The EBRS is a foundations belief revision system that attacks the minimal change problem. Like other RMS's, the EBRS receives new beliefs and justifications from an application system, accommodating them into the belief base. It then propagates the new information in the belief base. If a contradiction is identified, it retracts a set of beliefs to remove the contradictions. (The contradictions handled by the current implementation of the EBRS are *direct contradictions*, i.e., one belief is the negation of the other.[1]) The difference is that the EBRS ensures that the set being retracted (called the *obsolete belief set*) is minimal. A set of beliefs is a *candidate* of the obsolete belief set if retracting the set leads to removal of all contradictions from the belief base. A candidate is *minimal* if none of its proper subsets is also a candidate. The EBRS identifies the set of minimal candidates and selects the obsolete belief set from them. Following is a brief description of the EBRS. A full presentation can be

[1] The kinds of contradictions handled by different belief revision systems are varied. Propositional database updating systems detect and remove logical contradictions, while RMS's remove contradictions defined or detected by the application system. The EBRS does not impose the kind of contradiction to be removed, but in the current implementation, it detects and removes direct contradictions.

found in (Huang et al. 1991).

The EBRS contains a *modified ATMS* (de Kleer 1986), a *diagnostic system* and a *revision controller*. The ATMS is chosen from among many RMS's because it provides information about the minimal inconsistent belief spaces in which the direct contradictions hold. Using this information, the diagnostic system can find out the set of minimal candidates (Reiter 1987, de Kleer & Williams 1987). However, there might still be many minimal candidates. The *revision controller* is designed to select the obsolete set from the minimal candidates, based on *preference levels* associated to the beliefs in each minimal candidate.

The modified ATMS records beliefs in *EBRS nodes* which are similar to ATMS nodes and of the form: [statement, label, justifications]. The *statement* is a proposition that represents a belief. A *justification* is a constraint between the belief (called the *consequence*) and a set of existing beliefs (called the *antecedents*) used to derive the belief. The *label* is the set of minimal consistent *environments* in which the belief holds. An environment, representing a belief space, is a set of *base beliefs*. A base belief is a belief considered true by the system without depending on other beliefs. A base belief has no justification. Its label contains a singleton which contains the base belief itself. A contradiction (i.e., a direct contradiction) is recorded in a special kind of EBRS node called a *contradiction node*.

However, an ordinary ATMS is intended to be a temporary cache for a problem solver (de Kleer 1986). It is designed to support efficient search for solutions and fast switch of belief spaces. Whether the next belief space is close to the current one is not important. Thus, it has no notion of the current belief space. In contrast, the EBRS is used to maintain a belief base that reflects the current belief state of the application system (e.g., an intelligent tutor's beliefs about a student). It will not switch its belief spaces unless the current space has a contradiction. If it must switch the belief space (to revise the belief base), the next belief space should be as close to the current one as possible. The current belief space of the EBRS is represented by a set of base beliefs called the *current environment*. A proposition is believed if and only if the label of its EBRS node contains a subset of the current environment, or an *active environment*. Active environments in the labels of contradiction nodes are called *minimal conflicts*. They are inputs to the diagnostic system.

The diagnostic system uses a modified version of Reiter's (1987) HS-tree algorithm that returns all minimal hitting sets of a given set collection C.[2] Since in the EBRS C is the set of minimal conflicts, the algorithm returns all minimal candidates of the obsolete belief set. This ensures that the set of retracted beliefs is minimal.

[2] Given a set collection C = {Si | i = 1, ..., n}, where each Si is a set, a hitting set H for C is a set that contains at least one element of each set in C (Garey & Johnson 1979).

Some belief revision systems, such as Doyle's (1979) TMS and Martins and Shapiro's (1988) MBR, retract a base belief of each active environment of a contradiction node from the current environment. This may retract more beliefs than necessary. Consider a belief base containing base beliefs A, B, C and D, derived belief S, and justifications $A \wedge B \supset S$, $B \wedge C \supset S$ and $C \wedge D \supset S$. Now if a base belief $\neg S$ is obtained, then a contradiction node whose label contains the three active environments below would be created to record contradiction (S, $\neg$ S):

E1: $\{A, B, \neg S\}$; E2: $\{B, C, \neg S\}$;
E3: $\{C, D, \neg S\}$;

If one belief of each active environment is retracted from the current environment, for example, A (in E1), B (in E2), and C (in E3), then the revision is not minimal. A minimal candidate need only contain A and C.

The current implementation of the revision controller uses two preference levels in terms of the "primacy of new information" principle (Dalal 1988), that is, new beliefs are more preferable to retain in the belief base. Note that even if preference levels are carefully designed, there might still be situations where more than one minimal candidate is at the lowest level. In these situations, other information, such as next measurements discussed in (de Kleer & Williams 1987), must be employed to determine the obsolete set. Our current implementation invokes the application system to make this final decision.

3. Attention and Inconsistent Beliefs

Although the EBRS ensures minimality of belief revision, it requires intractable computation. The EBRS is built on the top of ATMS, and as in ATMS its nodes may have labels of exponential sizes (McAllester 1990). This inefficiency could be critical in an interactive application system with a large belief base. Motivated by the need to maintain individualized student models in intelligent tutoring (Huang et al. 1991, Wenger 1987), we have developed additional mechanisms to focus belief revision. These mechanisms are combined with the EBRS to form the ABRS mentioned in Section 1.

The ABRS deals with a large belief base that reflects the current belief state of the application system (called the *application* for short). In such a belief base, there is usually much information irrelevant to the problem that the application is currently dealing with. The basic idea of the ABRS is that revisions only change the subset of most relevant beliefs that are under the system's *attention*. The ABRS adds new beliefs to and maintains consistency for only this subset. But if the application moves to deal with another problem, the ABRS can replace the beliefs under attention by another subset of beliefs relevant to the new problem. Thus, the computation required for belief revision is only exponential to the size of the subset of beliefs under attention. If this subset is always restricted to be small enough, belief revision can be done quickly. Such a belief revision model may be analogous to people's belief revision. People can revise their beliefs quickly. It is argued that this efficiency comes from people's ability to focus on a small subset of beliefs in their working memory (Cherniak 1986). Although in this paper we have no intention to settle the psychological argument of how people revise beliefs, we will show how to use attention in a belief revision system to increase efficiency.

The trade off for this efficiency is that the global belief base of the system may not be consistent. But in many applications such as student modeling and user modeling in dialog systems (Kobsa & Wahlster 1989), if local consistency can be achieved, global consistency of the belief base is usually not compulsory (although it is preferable), whereas being able to respond promptly is crucial. It is widely agreed that people's beliefs are also inconsistent. Much AI research has been done on building computational models for inconsistent beliefs (Levesque 1984, Zadrozny 1986). In particular, Fagin and Halpern's (1988) *logic of local reasoning* views an intelligent agent as a *mind society* which contains many independent frames of mind. Although beliefs in each frame of mind are consistent, beliefs in different frames of mind may not be so. The ABRS is compatible with the mind society model, but we extend the model by distinguishing the frame of mind under attention. Also, we model changing beliefs instead of static beliefs. On the other hand, the ABRS is less formal than the logic of local reasoning. We aim at building an efficient belief revision system rather than investigating formal aspects of inconsistent beliefs.

4. The WM/LTM Architecture and Frames of Mind

The ABRS has two belief bases. The *working memory (WM)* holds beliefs under attention, and the *long-term memory (LTM)* stores other beliefs. This WM/LTM architecture is shown in Figure 1. The application obtains new beliefs by making observations and inferences. The ABRS provides two revision operations for the application to add new information into the WM: ADD-OBSERVATION and ADD-DERIVATION. The former responds to the situation that the application has just made an observation. It adds the set of obtained base beliefs to the WM. The latter responds to the situation that the application has just made an inference that results in a new belief. It adds the belief and a justification supporting the belief to the WM. Both operations call the EBRS to revise the WM so that the new beliefs are accommodated. Since the WM is small, belief revision can be done quickly. (Experiments show that it normally takes less than two seconds if the EBRS has 50 nodes.)

The LTM stores beliefs accepted (by the ACCEPT operation discussed in the next section) from the WM at the end of each *revision interval*. (A revision interval is a period between two calls of ACCEPT. Within each revision interval there are usually several *revision sessions*, namely calls of ADD-OBSERVATION or ADD-DERIVATION.) An LTM node is similar to a node in Doyle's TMS. It is of the form [statement, justifications, beliefp], where the "beliefp" entry is a predicate that indicates the belief status of the node (similar to "in" and "out" in Doyle's TMS).

The application can switch its attention to a new problem only if it can retrieve the set of beliefs relevant to the new problem into the WM. However, it would be very expensive if the application had to identify each relevant belief in a retrieval operation. To support fast determination of relevant beliefs, the ABRS uses *frames (of mind)* to pack relevant beliefs together. The LTM is covered by many frames. Each frame contains a set of relevant beliefs. Since a belief may be relevant to several problems, it may belong to more than one frame. Note that "frame" is a general notion. Depending on the application, a frame can be a *script* in a dynamic memory (Schank 1977), a *space* in a partitioned network (Hendrix 1979), a *solution element* in a blackboard system (Hayes-Roth 1987), or a *viewpoint* in an intelligent tutoring system (Self 1990), etc. We do not commit our frames to any specific application. From the view point of the ABRS, a frame, no matter what it is in the application, is simply a set of beliefs.

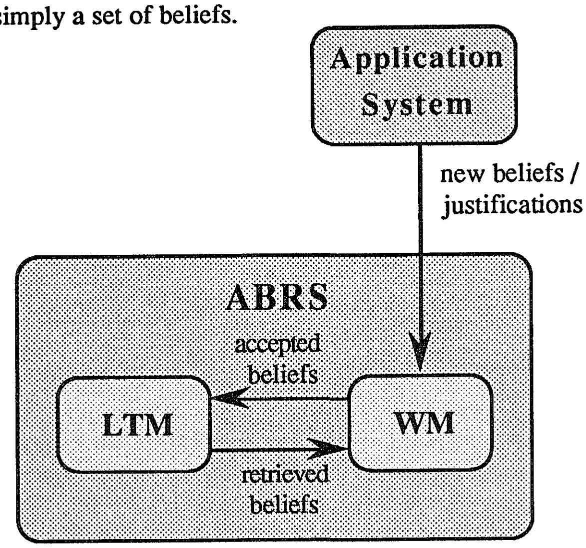

Figure 1. The Architecture of the ABRS

5. ACCEPT and RETRIEVE

The ABRS provides two operations, ACCEPT and RETRIEVE, to transfer information between the WM and the LTM and to support attention shifting. The ACCEPT operation accepts *WM beliefs* (propositions currently believed in the EBRS) and *valid justifications* (a justification is valid if all its antecedents are currently believed) to the LTM, and then clears the EBRS. For each new belief or new justification (a belief/justification not existing in the LTM) being accepted, the ABRS creates an LTM node/justification to record it.

ACCEPT also updates the belief status of LTM beliefs. If a disbelieved LTM node has a correspondent in the WM, the operation would change its belief status to "belief", carrying out a *belief propagation* in the LTM. On the other hand, if a belief retrieved from the LTM is represented by an EBRS node currently not believed, then ACCEPT would disbelieve the corresponding LTM node, carrying out a *disbelief propagation* in the LTM. If a belief being accepted was from or confirmed by an observation in the current revision interval, ACCEPT would mark the corresponding LTM node "base belief". An LTM base belief may not be disbelieved by a disbelief propagation because it is self justified, but it would be disbelieved if its WM correspondent is disbelieved, because a new observation can override an old observation (base beliefs are obtained from observations). A belief propagation or a disbelief propagation visits each LTM node at most once. Thus, the complexity of the ACCEPT operation is $O(MN)$, where M is the size of the LTM, and N is the size of the EBRS. It is not clear whether people also do such unconscious belief/disbelief propagation in their long-term memories. But since it is computationally cheap, the ABRS does it to update and to improve consistency of the LTM.

The RETRIEVE operation retrieves a frame indicated by the application from the LTM to the WM, enabling attention shifting. For example, in intelligent tutoring, the tutoring system may indicate retrieval of the frame(s) relevant to the topic that it is going to discuss with the student (Brecht 1990). In a dialog system, when a subtask is activated or completed, the focus space shifts (Grosz 1977). Then frame(s) associated with the new focus space may be retrieved. A frame being retrieved includes a set of beliefs and the justifications among them. Among the retrieved beliefs, base beliefs in the LTM and beliefs having no justification in the frame are treated as WM base beliefs. The intuition behind this decision is that the former are beliefs obtained from previous observations, and the latter are beliefs assumed to be true by the system without depending on other beliefs (since their justifications are not under attention).

Since the LTM may not be consistent, there may be contradictions among the retrieved beliefs. A retrieved belief may also contradict existing WM beliefs. Similar to ADD-OBSERVATION and ADD-DERIVATION, RETRIEVE calls the EBRS to remove contradictions after adding retrieved beliefs to the WM. The contradictions thus removed from the WM will also eventually be removed from the LTM when the WM beliefs are accepted to the LTM.

Consistency of the LTM may be further improved by a *sweep* process following an ACCEPT operation. The sweep process retrieves into the WM each frame changed by the ACCEPT, removing contradictions and then accepting the frame back to the LTM. This may in turn change other frames. The sweep process continues until all changed frames are consistent. Thus, if the LTM is locally consistent (in each frame) before the first ACCEPT operation, then the sweep process maintains this local consistency, accommodating information of the WM into the LTM. Since consistency maintenance only changes beliefs to disbeliefs but not the other way, the sweep process terminates in time $O(M^2 F)$, where F is the number of frames in the LTM.

6. Attention-Shifting Belief Revision

This section uses an example to show how the ABRS realizes attention-shifting belief revision. It also shows how consistency of the LTM is improved during the revision. In general, the ABRS assists the application to revise the belief base and to shift attention. When the application acts on the world, it obtains new beliefs and justifications by making observations and inferences. Then ADD-OBSERVATION and ADD-DERIVATION are applied to add the new information into the WM and to remove obsolete beliefs conflicting with the new information. If the application moves to deal with another problem, the ABRS would help the application to switch attention. Attention switching is accomplished in two steps. First, ACCEPT is applied to accept WM beliefs to the LTM. Then, RETRIEVE is used to retrieve the relevant frame to the WM.

The example is depicted in Figure 2, where small circles are LTM nodes, while small boxes are justifications. Among the LTM nodes, base beliefs are represented by solid circles, and disbeliefs are represented by crossed circles. Frames are represented by dashed big boxes. Figure 2 (a) shows the LTM before running the example. The LTM contains beliefs A, B, $\neg$B, C, D, E, U, V, W and X. Among these beliefs A, B, C, D, E and X are base beliefs. The LTM beliefs are covered by three frames: F1, F2 and F3. A, B, C and U are in F1. $\neg$B, D, E, V and X are in F2. $\neg$B, U, V and W are in F3. Note that some beliefs belong to more than one frame, and that there are unnoticed contradictions in the LTM.

Suppose that the application is paying attention to frame F1, so beliefs in F1 are retrieved to the WM (the operation: (RETRIEVE F1 E-EBRS)). No contradiction is discovered. Then the application makes an observation, obtaining new belief $\neg$U (the operation: (ADD-OBSERVATION '($\neg$U) E-EBRS)). This brings a contradiction, (U, $\neg$U), to the WM. In order to remove the contradiction, the EBRS must retract U, which further requires the retraction of a minimal set of WM base beliefs supporting U. There are two candidates for the minimal set: A and B, C. Suppose that A is chosen, so A and U are removed from the WM. The application then applies an inference rule $B \wedge \neg U \supset Y$ to derive Y (the operation: (ADD-DERIVATION '(B $\neg$U) 'Y E-EBRS)). Now the WM contains four beliefs: B, C, $\neg$U and Y. Also, two retrieved beliefs, A and U, have been disbelieved.

Suppose that at this time the application moves to work on another problem to which frame F3 is relevant. The ABRS helps it to switch attention. First, ACCEPT is applied (the operation: (ACCEPT E-EBRS)). It accepts the four WM beliefs to the LTM, creating an LTM node to record new belief Y. It also disbelieves two LTM nodes, A and U, because their WM correspondents were disbelieved. Disbelief propagation in the LTM changes belief status of node W to "disbelief". Figure 2 (b) shows the updated LTM. (The WM is cleared at the end of the operation.)

Then F3 is retrieved to the empty WM. U and W are disbeliefs, so the WM contains only V and $\neg$B. This completes the attention switching. The application now can use beliefs relevant to the new problem. The succeeding revision operations manipulate new contents of

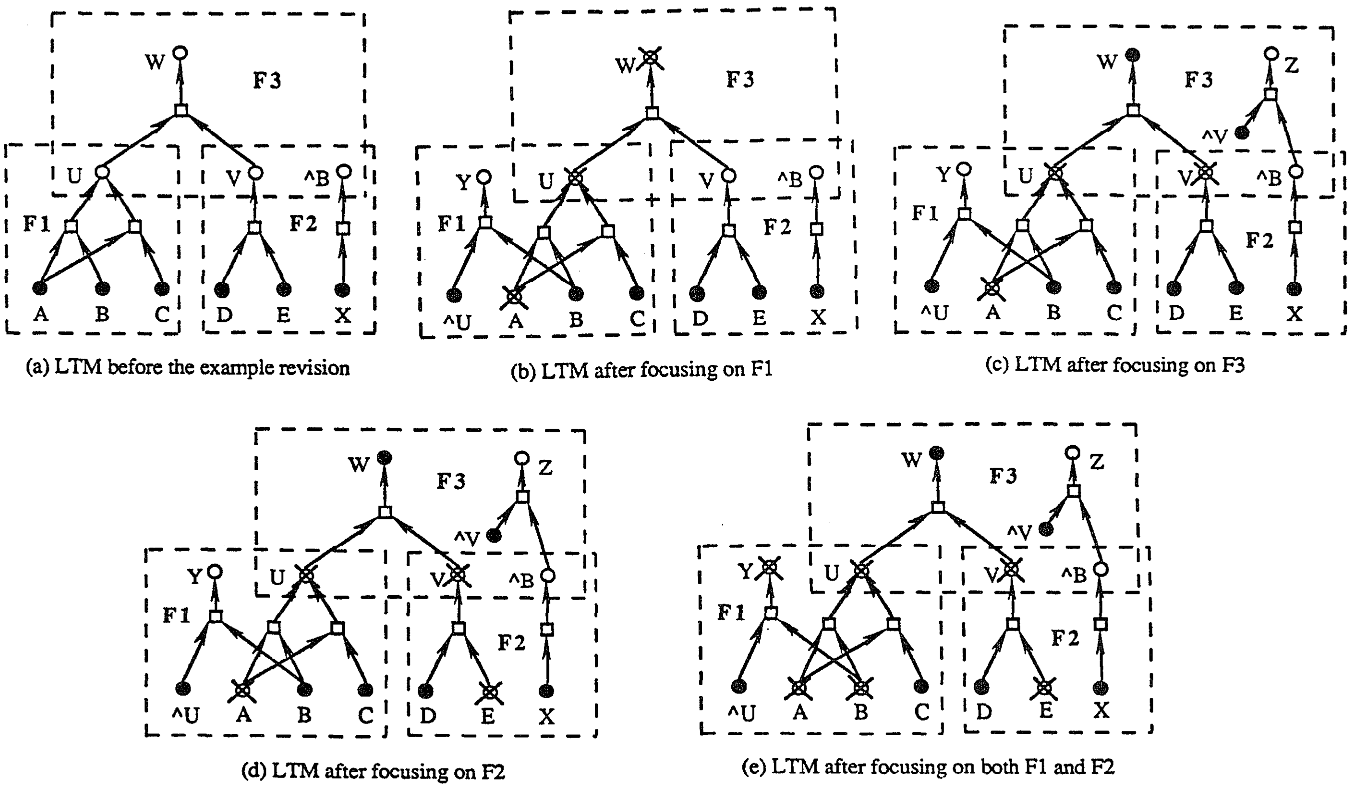

Figure 2. An Example of Attention-Shifting Belief Revision

the WM. Suppose that in this new revision interval, an observation and an inference is made. The observation obtains two beliefs, $\neg$ V and W, and the inference uses a rule $\neg B \wedge \neg V \supset Z$ to generate belief Z. These new beliefs are added to the WM, causing V to be disbelieved. Note that W was a disbelief, but now is re-believed because it gains a new support from the observation. If now ACCEPT is applied, then the LTM would be revised again as shown in Figure 2 (c).

In Figure 2 (c), a constraint in the LTM, $D \wedge E \supset V$ is violated. The ABRS is a "lazy" reviser (its efficiency derives partly from this laziness). It does not remove the inconsistency until all antecedents and the consequence of the constraint are in the WM. This can be done by retrieving F2 into the WM. The EBRS discovers the inconsistency, so it disbelieves E (alternatively, the EBRS could disbelieve D instead) to remove the inconsistency. Now we accept the WM beliefs. E is disbelieved in the LTM, and the inconsistency is removed (Figure 2 (d)). Note that retrieval of F2 and removal of its inconsistency would be done automatically without interacting with the application if a sweep process follows the ACCEPT operation for F3.

The ABRS may also retrieve more than one frame into the WM, remove contradictions and relax constraints between the retrieved frames, and then accept changes back to the LTM. This can improve consistency of the LTM (although inconsistencies may still remain). For example, by using (RETRIEVE F1 E-EBRS) and then (RETRIEVE F2 E-EBRS), F1 and F2 are retrieved into the WM. The EBRS discovers a contradiction (B, $\neg$ B). It must retract either base belief B and its consequence or base belief X and its consequence from the WM to remove the contradiction. Suppose that it retracts B and its consequence Y. Then let the ABRS accept the WM beliefs. This results in the LTM depicted in Figure 2 (e) in which the contradiction is removed.

7. Summary and Discussion

We have attacked the efficiency problem in belief revision by localizing revisions to the WM. The EBRS maintains consistency of the WM by making minimal revision to it. With ACCEPT and RETRIEVE, the ABRS enables the application to switch attention, which allows use and revision of different subsets of beliefs in the LTM. In addition, the ABRS supports efficient retrieval of relevant beliefs by indexing them using frames.

The ABRS and the EBRS have been implemented in Allegro Common Lisp on a Sun-4 workstation. We have run both systems on four belief bases with different sizes. The experimental results show that the time for belief revision with the ABRS is indeed independent of the size of the belief base (the LTM), and as expected, the time for the ACCEPT operation, the RETRIEVE operation and the sweep process increase gradually as the size of the LTM increases. In contrast, the time for belief revision with the

EBRS, which has no attention focusing, increases drastically, supporting the prediction of exponential growth. Details of the empirical experiments are described in (Huang 1991).

The efficiency of the ABRS is gained at the cost of global consistency in the LTM. Thus, it is advisable not to use the ABRS for applications that require a consistent global belief base, but rather applications in which global consistency is not compulsory. In any event, the ABRS improves consistency in the LTM by doing (forward) constraint propagation in it. It can also efficiently maintain local consistency for all frames in the LTM by running the sweep process. If consistency in several frames of the LTM is necessary, these frames could be retrieved to the WM. Then the ABRS could call the EBRS to remove contradictions and put back the revised beliefs to the LTM.

de Kleer (1990) also discusses locality in truth maintenance, but his work emphasizes improving efficiency of a logically complete TMS, while the ABRS is designed for efficiently updating a belief base. The two systems use very different approaches and techniques.

The current implementation assumes that each potential belief is pre-assigned to some frames when the ABRS is initialized. This may work well for some applications such as student modeling where we can pre-assign concepts and likely misconceptions in a sub-domain of knowledge to a particular frame. But dynamic assignment of beliefs to frames is also possible and may be more suitable for some other applications. For example, a new belief could be assigned to the frames currently in the WM. What frame assignment strategy is used depends on the application. The ABRS does not commit to any specific strategy.

This paper has shown that attention shifting is useful for belief revision. We expect that it is also useful for other AI problems since it seems to be a way that people overcome the complexity in reasoning (Cherniak 1986).

Acknowledgements

Thanks to the University of Saskatchewan, the Natural Sciences and Engineering Research Council and the IRIS project for their financial support.

Reference

Brecht, B. J., 1990. *Determining the Focus of Instruction: Content Planning for Intelligent Tutoring Systems*, Research Report 90-5, ARIES Laboratory, Dept. of Computational Science, Univ. of Saskatchewan.

Cherniak, C., 1986. *Minimal Rationality*, The MIT Press.

Dalal, M., 1988. Investigation into a theory of knowledge base revision: preliminary report, *Proceedings AAAI-88*, Saint Paul, 475-479.

de Kleer, J., 1986. An assumption-based TMS, *Artificial Intelligence* **28** (2), 127-162.

de Kleer, J., 1990. Exploiting locality in a TMS, *Proceedings AAAI-90*, Boston, 264-271.

de Kleer, J. and Williams, B. C., 1987. Diagnosing multiple faults, *Artificial Intelligence* **32** (1), 97-130.

Doyle, J., 1979. A truth maintenance system, *Artificial Intelligence* **12**, 231-272.

Fagin, R. and Halpern, J. Y., 1988. Belief, awareness, and limited reasoning, *Artificial Intelligence* **34** (1), 39-76.

Gardenfors, P., 1990. The dynamics of belief systems: foundations vs. coherence theories, *Revue Internationale de Philosophie* .

Garey, M. R. and Johnson, D. S., 1979. *Computers and Intractability: A Guide to the Theory of NP-Completeness*, W. H. Freeman and Company.

Grosz, B. J., 1977. The representation and use of focus in a system for understanding dialog, *Proceedings IJCAI-77*, Cambridge, 67-76.

Hayes-Roth, B., 1987. Blackboard systems, in Shapiro, S. C. (ed.) *Encyclopedia of Artificial Intelligence*, Wiley-Interscience Publications, 73 - 80.

Hendrix, G. G., 1979. Encoding knowledge in partitioned networks, in Findler, N. V. (ed.), *Associative Network*, Academic Press, 51-92.

Huang, X., 1991. *Updating Belief Systems*, Ph. D. Dissertation, Dept. of Computational Science, Univ. of Saskatchewan (in preparation).

Huang, X., McCalla, G. I., Greer, J. E. and Neufeld, E., 1991. Revising deductive knowledge and stereotypical knowledge in a student model, *User Modeling and User-Adapted Interaction* **1** (1) (in press).

Katsuno, H. and Mendelzon, A. O., 1989. A unified view of propositional knowledge base updates, *Proceedings IJCAI-89*, Detroit, 1413-1419.

Kobsa, A and Wahlster, W. (eds.), 1989. *User Modeling in Dialog Systems*, Springer-Verlag.

Levesque, H. J., 1984. A logic of implicit and explicit belief, *Proceedings AAAI-84*, Austin, 198-202.

Martins, J. P. and Shapiro, S. C., 1988. A model for belief revision, *Artificial Intelligence* **35** (1), 25-79.

McAllester, D. A., 1990. Truth maintenance, *Proceedings AAAI-90*, Boston, 1109-1115.

Rao, A. S. and Foo, N. Y., 1989. Formal theories of belief revision, *Proceedings of the First International Conference on Principles of Knowledge Representation and Reasoning*, Toronto, 369-380.

Reiter, R., 1987. A theory of diagnosis from first principles, *Artificial Intelligence* **32** (1), 57-95.

Schank, R. C. and Abelson, R. P., 1977. *Scripts, Plans, Goals and Understanding*, Lawrence Erlbaum, Hillsdale.

Self, J., 1990. *Computational Viewpoints*, AI-Report No. 44, Dept. of Computing, Univ. of Lancaster.

Weber, A., 1986. Updating propositional formulas, *Proceedings of the First International Conference on Expert Database Systems* , Charleston, 487-500.

Wenger, E., 1987. *Artificial Intelligence and Tutoring Systems*, Morgan Kaufman Publishers, Inc.

Zadrozny, W., 1986. Explicit and implicit beliefs, *Proceedings of the Tenth Workshop on Artificial Intelligence*.