Sponsor: International Joint Conferences on Artificial Intelligence, Inc.

Cosponsors: Gesellschaft für Informatik (GI)
Society for the Study of Artificial Intelligence and the Simulation of Behaviour (AISB)
Nederlandse Vereniging voor Kunstmatige Intelligentie (NVKI)

Industrial Cosponsors: British Telecom
Cambridge Consultants Limited
GEC Research Laboratories
ITT Europe Engineering Support Centre
Shell Research Limited

ijcai-83

Proceedings of the Eighth International Joint Conference on Artificial Intelligence

8-12 August 1983, Karlsruhe, West Germany

Volume 1

REASONING IN MULTIPLE BELIEF SPACES

João P. Martins* and Stuart C. Shapiro

Department of Computer Science
State University of New York at Buffalo
Amherst, N.Y. 14226, U.S.A.

ABSTRACT

MBR is a reasoning system which allows multiple beliefs (beliefs from multiple agents, contradictory beliefs, hypothetical beliefs) to be represented simultaneously in the same knowledge base and performs reasoning within sets of these beliefs. MBR also contains provisos to detect contradictions and to recover from them.

This paper describes MBR's method of detecting and recording contradictions within beliefs of different agents, showing an example of such process.

1. Introduction

This paper reports a small feature of a large system, the MBR (Multiple Belief Reasoner) system [3]. MBR is fully implemented in Franz Lisp, running on a VAX-11/750. MBR is a reasoning system which allows multiple beliefs (beliefs from multiple agents, contradictory beliefs, hypothetical beliefs) to be represented simultaneously in the same knowledge base and performs reasoning within sets of these sets of beliefs. MBR also contains provisos for detecting contradictions and for recovering from them.

The problem of detecting and recording contradictions has been considered by several researchers (e.g., [2, 4, 5]). The part of MBR that deals with this problem differs from the previous approaches because, 1) It is based on a logic developed for such purpose; 2) It is implemented such that the detection of the hypotheses underlying the contradiction is done by following only two types of arcs; there is no need to explicitly mark propositions as believed or disbelieved; there is no need to worry about circular proofs; there is no need to keep a separate data structure to record previous contradictions.

2. The SWM system

The SWM system is the logical system underlying MBR. It is loosely based on the logical systems of [1] and [7]. Distinguishing features of SWM include recording dependencies of wffs, not allowing irrelevancies to be introduced, and providing for dealing with contradictions.

The SWM system deals with objects called supported wffs which are of the form $F:\tau,\alpha,\rho$, in which F is a wff, τ (the origin tag) is an element of the set {hyp, der, ext}, α (the origin set) is a set of hypotheses, and ρ (the restriction set) is a set of sets of hypotheses. The origin tag (OT) tells whether F is an hypotheses (τ=hyp), a normally derived wff (τ=der) or a wff with an extended OS (τ=ext) (this latter case will not be discussed in this paper). The origin set (OS) contains all the hypotheses which were actually used in the derivation of F. The restriction set (RS) contains sets of hypotheses, each of which when unioned with the hypotheses in the OS forms a set which is known to be inconsistent. An inconsistent set, is a set of wffs from which a contradiction may be derived.

RSs are very different entities from OTs and OSs. Whereas the OT and OS of a proposition reflect the way the proposition was derived, the RS of a proposition reflects the current knowledge about how the hypotheses underlying that proposition relate to the other hypotheses in the system. Once a proposition is derived its OT and OS remain constant.

---

* Current address: Departamento de Engenharia Mecanica, Instituto Superior Tecnico, Av. Rovisco Pais, 1000 Lisboa, Portugal

This work was supported in part by the National Science Foundation under Grant MCS80-06314 and by the Instituto Nacional de Investigação Cientifica (Portugal) under Grant No.20536.

whereas its RS changes as new inconsistencies are uncovered in the system.

3. Contexts and Belief Spaces

MBR is to be used as the deduction system in a knowledge base which may contain information entered by many users, with different and even conflicting interests. We assume that each user of the knowledge base has some basic set of beliefs which he/she told MBR about. Such beliefs are the user's basic assumptions and were entered into the knowledge base as hypotheses. Every proposition derived from this set of assumptions is assumed to be be believed by the user.

We define a context to be a set of hypotheses. A context represents the set of assumptions of some user. A context determines a Belief Space (BS) which is the set of all the hypotheses defining the context and all the propositions which were derived from them. Within the SWM formalism (the logic underlying MBR), the propositions in a given BS are characterized by having an OS which is contained in the context.

At any point, the set of all hypotheses believed is termed the current context (CC), which defines the current belief space (CBS).

Contexts delimit smaller knowledge bases (called Belief Spaces) within the knowledge base. The knowledge base retrieval operations only retrieve the propositions within the CBS, ignoring all other propositions.

4. The handling of contradictions

In this section, we will discuss how contradictions are handled by MBR.

MBR relies on the two rules of inference of SWM that handle contradictions: negation introduction (~I) and updating of restriction sets (URS).

The rule of ~I states that from $A:\tau,\alpha,\rho$ and $\sim A:\zeta,\beta,\theta$ (meaning that from the hypotheses in $\alpha\cup\beta$ a contradiction, A&~A, can be derived) we can deduce the negation of the conjunction of any number of hypotheses in $\alpha\cup\beta$ under an OS containing the remaining hypotheses. The meaning of this newly derived wff is that such conjunction can not be considered under the assumption of the other hypotheses.

The rule of URS states that from $A:\tau,\alpha,\rho$ and $\sim A:\zeta,\beta,\theta$ we must update the RS of every hypothesis in $\tau\cup\zeta$ and of all the wffs derived from them. This updating has the effect of recording the existence of the inconsistent set $\alpha\cup\beta$. Details of how this is done can be found in [3].

Based in these two rules of inference, whenever MBR finds a contradiction it takes one of the following actions:

1. If only one of the contradictory wffs belongs to the CBS the contradiction is recorded (through the application of URS) but nothing more happens. The effect of doing so is to record that some set of hypotheses, strictly containing the CC, is now known to be inconsistent.
2. If both contradictory wffs belong to the CBS. Then the rule of URS is applied but, in addition, the rule of ~I is also applied. This has the effect of adding new wffs to the knowledge base and also will cause the CC to be revised.

5. An Annotated Example

We present in this section a sample run using MBR. Suppose that MBR is being used by some university as a meeting scheduling system. The knowledge base contains, in this case, general statements reflecting policies for scheduling meetings and also statements concerning the particular schedules of the users of the system.

MBR is asked to schedule meetings among a certain number of its users and it does so either by finding a time slot which is compatible with their particular schedules or by reporting that the schedules of the users do not allow the scheduling of the desired meeting. In this example we will assume that:

1. Meetings are being scheduled within one day only, therefore information about dates is absent from our representation;
2. Meetings can not both be in the morning and in the afternoon (hyp1, Fig.1).
3. Two different meetings can not fill the same time slot, i.e., morning or afternoon (hyp2, Fig.1).

We will follow MBR's behavior using the information contained in the schedules of two of its users, Stu and Tony. Both Stu and Tony already have some scheduled meetings:

1. Stu's schedule: Stu teaches a seminar in the morning (hyp6, Fig.1).
2. Tony's schedule: Tony has a tennis

game sheduled for the afternoon (hyp7, Fig.1).

The knowledge base also contains information about which objects are meetings (hyp3, hyp4 and hyp5, Fig.1). Figure 1 shows the knowledge base for this small example. As a shorthand, we do not represent OS's as sets of hypotheses but rather as sets of mnemonics representing the hypotheses.

```
hyp1 : ∀(x)(meeting(x) →
               (time(x,morning) →
                 ~time(x,afternoon))):hyp,(hyp1),()

hyp2 : ∀(x,y)((meeting(x) & meeting(y) & x≠y) →
               ((time(x,morning) → time(y,afternoon))
                &
                (time(x,afternoon) → time(y,morning))))
             :hyp,(hyp2),()

hyp3 : meeting(seminar):hyp,(hyp3),()

hyp4 : meeting(tennis-game):hyp,(hyp4),()

hyp5 : meeting(faculty-meet):hyp,(hyp5),()

hyp6 : time(seminar,morning):hyp,(hyp6),()

hyp7 : time(tennis-game,afternoon):hyp,(hyp7),()
```

figure 1
Hypotheses in the knowledge base

Suppose that Stu wants to schedule a faculty meeting and he wants to do so according to his schedule: he considers the general statements about meetings (hyp1, hyp2, hyp3 and hyp5) and also considers the statements that reflect his schedule (hyp6). In other words he does reasoning within the BS defined by the context Stu-schedule=(hyp1, hyp2, hyp3, hyp5, hyp6). Within this BS, MBR derives the wffs represented in Figure 2. After

```
wff1:  time(seminar,morning) → time(faculty-meet,afternoon)
                                   :der,(hyp2,hyp3,hyp5),()

wff2:  time(faculty-meet,afternoon):der,(hyp2,hyp3,hyp5,hyp6),()

wff3:  time(faculty-meet,morning) →
          ~time(faculty-meet,afternoon) :der,(hyp1,hyp5),()

wff4:  ~time(faculty-meet,morning)
                           :der,(hyp1,hyp2,hyp3,hyp5,hyp6),()
```

figure 2
wffs derived from "Stu-schedule"

this session Stu concludes that the best time, for him, for scheduling the faculty meeting is in the afternoon (wff2).

Suppose now that Tony also tries to find the most convenient time, for him, to have a faculty meeting. In this case, he does reasoning in the BS defined by the context Tony-schedule=(hyp1, hyp2, hyp4, hyp5, hyp7). Some results of such

```
wff5:  time(tennis-game,afternoon) →
          time(faculty-meet,morning):der,(hyp2,hyp4,hyp5),()

wff6:  time(faculty-meet,morning):der,(hyp2,hyp4,hyp5,hyp7),()
```

figure 3
wffs derived from "Tony-schedule"

reasoning are represented in Figure 3. In this case, however, when wff6 (time(faculty-meet,morning)) is derived the system finds out that it contradicts wff4 (~time(faculty-meet,morning)). The implementation of MBR guarantees that this wff is found without having to search the entire knowledge base. Since wff4 does not belong to the CBS (which is now the BS defined by the context "Tony-schedule") there is no visible contradiction. However the system records that the union of the contexts "Stu-schedule" and "Tony-schedule" ((hyp1, hyp2, hyp3, hyp4, hyp5, hyp6, hyp7)) is an inconsistent context. The rule URS is applied resulting in the knowledge base represented in Figure 4. The system

```
hyp1: ∀(x)(meeting(x) →
            (time(x,morning) → ~time(x,afternoon)))
          :hyp,(hyp1),((hyp2,hyp3,hyp4,hyp5,hyp6,hyp7))

hyp2: ∀(x,y)((meeting(x) & meeting(y) & x≠y) →
          ((time(x,morning) → time(y,afternoon))
           &
           (time(x,afternoon) → time(x,morning))));
            :hyp,(hyp2),((hyp1,hyp3,hyp4,hyp5,hyp6,hyp7))

hyp3: meeting(seminar)
            :hyp,(hyp3),((hyp1,hyp2,hyp4,hyp5,hyp6,hyp7))

hyp4: meeting(tennis-game)
            :hyp,(hyp4),((hyp1,hyp2,hyp3,hyp5,hyp6,hyp7))

hyp5: meeting(faculty-meet)
            :hyp,(hyp5),((hyp1,hyp2,hyp3,hyp4,hyp6,hyp7))

hyp6: time(seminar,morning)
            :hyp,(hyp6),((hyp1,hyp2,hyp3,hyp4,hyp5,hyp7))

hyp7: time(tennis-game,afternoon)
            :hyp,(hyp7),((hyp1,hyp2,hyp3,hyp4,hyp5,hyp6))

wff1: time(seminar,morning) → time(faculty-meet,afternoon)
            :der,(hyp2,hyp3,hyp4),((hyp1,hyp5,hyp6,hyp7))

wff2: time(faculty-meet,afternoon)
            :der,(hyp2,hyp3,hyp5,hyp6),((hyp1,hyp4,hyp7))

wff3: time(faculty-meet,afternoon) → ~time(faculty-meet,morning)
            :der,(hyp1,hyp5),((hyp2,hyp3,hyp4,hyp6,hyp7))

wff4: ~time(faculty-meet,morning)
            :der,(hyp1,hyp2,hyp3,hyp5,hyp6),((hyp4,hyp7))

wff5: time(tennis-game,afternoon) → time(faculty-meet,morning)
            :der,(hyp2,hyp4,hyp5),((hyp1,hyp3,hyp6,hyp7))

wff6: time(faculty-meet,morning)
            :der,(hyp2,hyp4,hyp5,hyp7),((hyp1,hyp3,hyp6))
```

figure 4
knowledge base after URS

reports to Tony that the faculty meeting should be in the afternoon.

Suppose that someone now wants to schedule a faculty meeting with all the members of the faculty, which include both Stu and Tony. When that request is made considering a context containing "Stu-schedule" and "Tony-schedule" the system immediately reports that such context is inconsistent. Notice that this context contains, possibly among other hypotheses, the hypotheses hyp1, hyp2, hyp3, hyp4, hyp5, hyp6 and hyp7. The RS of hyp1, for example, is ((hyp2, hyp3, hyp4, hyp5, hyp6, hyp7)) (Figure 4), which records that the set of hypotheses hyp1 through hyp7 is inconsistent. The system responds that such context is inconsistent and a revision of the CC should be performed.

Suppose now that starting from the knowledge base represented in Figure 1 the request is made to schedule the faculty meeting in a BS defined by a context containing "Stu-schedule" and "Tony-schedule".

In this case, there are no recorded inconsistencies and the system will try to schedule the faculty meeting in that BS. Among the results derived are the wffs represented in Figure 5. In this case,

```
wff1':  ~time(faculty-meet,morning)
                              :der,(hyp1,hyp2,hyp3,hyp5,hyp6),()

wff2':  time(faculty-meet,morning):der,(hyp2,hyp4,hyp5,hyp7),()
```

figure 5
wffs derived within the CC

both wff1' and wff2' belong to the CBS (the CC contains the hypotheses hyp1, hyp2, hyp3, hyp4, hyp5, hyp6, hyp7). Therefore, not only the rule of URS is applied, recording the inconsistent set, but also ~I is applied in order to rule out some hypothesis (or hypotheses) defining the CC.

6. Concluding Remarks

MBR has been implemented in Franz Lisp (runing on a VAX-11/750) using the SNePS system [6]. The example presented here was obtained from an actual run just just by slightly changing the output syntax.

One of the main distinguishing characteristics of MBR is that it is based on a logic (SWM) especially designed for Belief Revision systems.

In MBR propositions are represented by network nodes and are linked with the hypotheses in their OS and the sets in their RS. This way of representing propositions makes it possible to know a priori the number of arcs that has to be traversed to find out all the hypotheses underlying a contradiction.

Another characteristic of MBR concerns the way contexts and BS are defined. By defining a context as a set of hypotheses we can have as many contexts in the system as the power set of the hypotheses introduced. Also, the network retrieval functions only consider the propositions in the CBS. When a contradiction is detected, after selecting one hypothesis (or several hypotheses) as the culprit for the contradiction, the disbelief in all the propositions depending on such hypothesis (hypotheses) is done just by dropping it (them) from the CC. From then on, all such propositions are disregarded by MBR.

Finally the definition of RSs waives the need to keep a separate data strucure to record all the previous contradictions (e.g., the NOGOOD list [2]).

ACKNOWLEDGEMENTS

Many thanks to Gerard Donlon, Donald McKay, Ernesto Morgado, Terry Nutter, Bill Rapaport and the other members of the SNePS Research Group for their comments and criticisms concerning the current work.

REFERENCES

[1] Anderson A. and Belnap N., Entailment: The Logic of Relevance and Necessity, Vol.1, Princeton University Press, 1975.

[2] Doyle J., "A Truth Maintenance System", Artificial Intelligence 12:3 (1979) 231-272.

[3] Martins J., "Reasoning in Multiple Belief Spaces", Ph.D. Dissertation, Department of Computer Science, SUNY at Buffalo, May 1983.

[4] McAllester D., "An Outlook on Truth Maintenance", A.I. Lab., M.I.T., AI Memo 551, 1980.

[5] McDermott D., "Contexts and Data Dependencies: A Synthesis", Department of Computer Science, Yale University, 1982.

[6] Shapiro S., "The SNePS semantic network processing system", in Associative Networks, N.V.Findler (ed.), Academic Press, pp.179-203, 1979.

[7] Shapiro S. and Wand M., "The Relevance of Relevance", Technical Report No.46, Computer Science Department, Indiana University, Indiana, 1976.