An Algorithm for Belief Revision

Renata Wassermann
Institute for Logic, Language and Computation
University of Amsterdam
Plantage Muidergracht 24
1018-TV, Amsterdam, The Netherlands
email: renata@wins.uva.nl

Abstract

In this paper we show that a particular construction of belief revision operator is equivalent to the standard method for computing consistency-based diagnosis. We show how a diagnosis problem can be translated into a problem of belief revision and show how kernel constructions for revision operators can be used for computing diagnosis. We also show how Reiter's algorithm for computing diagnosis can be adapted for being used in belief revision.

1 Introduction

Belief revision (for an overview, see [Gärdenfors, 1988; Gärdenfors and Rott, 1995]) deals with the problem of how to accommodate new assertions into an existent body of knowledge. Traditionally, the body of knowledge is represented by a belief set, a set of formulas closed under logical implication.

Instead of belief sets we are going to use belief bases to represent belief states. A belief base is a set not closed under logical consequence [Fuhrmann, 1991; Hansson, 1989; Nebel, 1992]. For every belief base B, its closure $Cn(B)$ is a belief set that represents the beliefs held by the agent. The elements of B are assumed to be in a sense more basic beliefs, from which the elements of $Cn(B) \setminus B$ are derived. Belief bases have substantial advantages in terms of computability [Nebel, 1998], and their increased expressive power as compared to belief sets can be used to represent important features of actual belief systems [Hansson, 1992].

In AGM theory [Alchourrón *et al.*, 1985], three forms of belief change are identified: contraction, expansion, and revision. Contraction consists of retracting a specified sentence from the belief set. Expansion consists of adding a specified sentence to the belief set. If the old and the new information are not logically compatible, then the new belief state after expansion will be inconsistent. Revision is consistency-preserving incorporation of new information, i.e. if the input sentence is consistent, then the new belief set will be consistent. If necessary, consistency is obtained by deleting parts of the original belief set. These operations have also been defined for belief bases [Fuhrmann, 1991; Hansson, 1991; Hansson, 1999b].

Two additional operations of change on belief bases were introduced in [Hansson, 1991] and [Hansson, 1997]: consolidation and semi-revision. Consolidation consists in making an inconsistent belief base consistent. Semi-revision is an operation that may either accept or reject the input sentence. (For corresponding operations on belief sets, see [Makinson, 1997] and [Hansson, 1999a]).

Expansion is the only of the five operations which does not involve any extra-logical information. Expanding a belief set K by α consists in taking the logical consequences of K together with α, i.e., $K + \alpha = Cn(K \cup \{\alpha\})$. The result of the expansion of a belief base B by α is simply the union of B and α, i.e., $B + \alpha = B \cup \{\alpha\}$.

The other four operations are not uniquely defined and have been characterized in the literature by means of rationality postulates and constructions. The main purpose of this paper is to show that one sort of construction found in the literature is directly related to the constructive approach given by Reiter for finding diagnoses in faulty systems [Reiter, 1987].

We will show how a diagnosis problem can be translated into an operation of kernel semi-revision. Kernel semi-revision consists in adding new information to a database and restoring consistency if necessary. To restore consistency, the expanded database is contracted by $\perp$.

We will show how the traditional algorithm for consistency-based diagnosis given by Reiter can be used for implementing semi-revision.

Winslett suggests the use of belief revision techniques for modeling diagnosis, but without analyzing the similarities between the constructions proposed in both fields [Winslett, 1995]. She only shows how a particular problem of diagnosis can be formalized as a belief revision problem. Nebel also points that syntax-based approaches to belief revision are appropriated for problems of diagnosis, but without exploring the similarity of the constructions [Nebel, 1998].

Beyond just reducing the diagnosis problem to a problem of belief revision, the present paper aims at opening a cross-fertilization process between two communities. Researchers working on belief revision rely on very elegant and precise logical formalisms, but are very far from implementing a realistic belief revisioner. On the other hand, researchers working in the field of diagnosis have very powerful tools to prune the computational complexity of the problem, allowing them to deal with real-world situations. But several applications lack a clear formalization. By showing that, at least at a high level, the problems are equivalent, we claim that techniques developed by the model-based diagnosis community could be used for implementing belief revision.

In the rest of this paper we consider L to be a propositional language closed under the usual truth-functional connectives and containing a constant $\perp$ denoting falsum.

2 Constructions for Belief Revision

Partial meet contraction, introduced in [Alchourrón *et al.*, 1985], uses a selection function to select some of the maximal subsets of a belief set (or belief base) which do not imply the formula to be contracted. The result of the contraction is the intersection of the selected subsets. There are two limiting cases: only one subset is selected or all subsets are selected. In the first case, the operation is called a maxichoice contraction and in the second, a full meet contraction.

Hansson introduced another construction for contraction operators, called *kernel contraction* [Hansson, 1994], which is a generalization of the operation of safe contraction defined in [Alchourrón and Makinson, 1985]. The idea behind kernel contraction is that, if we remove from the belief base B at least one element of each α-kernel (minimal subset of B that implies α), then we obtain a belief base that does not imply α [Hansson, 1994]. To perform these removals of elements, we use an incision function, i.e., a function that selects at least one sentence from each kernel.

Definition 2.1 *[Hansson, 1994] The* **kernel** *operation* $\perp\!\!\!\perp$ *is the operation such that for every set* B *of formulas and every formula* α, $X \in B \perp\!\!\!\perp \alpha$ *if and only if:*

1. $X \subseteq B$
2. $\alpha \in Cn(X)$
3. *for all* Y, *if* $Y \subset X$ *then* $\alpha \notin Cn(Y)$

The elements of $B \perp\!\!\!\perp \alpha$ *are called* α*-kernels.*

Definition 2.2 *[Hansson, 1994] An* **incision function** *for* B *is any function* σ *such that for any formula* α:

1. $\sigma(B \perp\!\!\!\perp \alpha) \subseteq \bigcup(B \perp\!\!\!\perp \alpha)$, *and*
2. *If* $\emptyset \neq X \in B \perp\!\!\!\perp \alpha$, *then* $X \cap \sigma(B \perp\!\!\!\perp \alpha) \neq \emptyset$.

Semi-revision consists of two steps: first the belief α is added to the base, and then the resulting base is consolidated, i.e., contracted by $\perp$.

Definition 2.3 *[Hansson, 1997] The* **kernel semi-revision** *of* B *based on an incision function* σ *is the operator* $?_\sigma$ *such that for all sentences* α:

$$B?_\sigma\alpha = (B \cup \{\alpha\}) \setminus \sigma((B \cup \{\alpha\}) \perp\!\!\!\perp \perp)$$

Theorem 2.4 *[Hansson, 1997] An operator* $?$ *is an operator of kernel semi-revision if and only if for all sets* B *of sentences:*

- $\perp \notin Cn(B?\alpha)$ *(consistency)*
- $B?\alpha \subseteq B \cup \{\alpha\}$ *(inclusion)*
- *If* $\beta \in B \setminus B?\alpha$, *then there is some* $B' \subseteq B \cup \{\alpha\}$ *such that* $\perp \notin Cn(B')$ *and* $\perp \in Cn(B' \cup \{\beta\})$ *(core-retainment)*
- $(B + \alpha)?\alpha = B?\alpha$ *(pre-expansion)*
- *If* $\alpha, \beta \in B$, *then* $B?\alpha = B?\beta$ *(internal exchange)*

Kernel operations are more general than partial meet, i.e., all partial meet operations can be obtained by kernel operations but the converse does not hold [Hansson, 1999b].

3 Consistency-Based Diagnosis

Diagnosis is a very active area within the artificial intelligence community. The problem of diagnosis consists in, given an observation of an abnormal behavior, finding the components of the system that may have caused the abnormality [Reiter, 1987].

In the area known as model-based diagnosis [Hamscher *et al.*, 1992], a model of the device to be diagnosed is given in some formal language. In this paper, we will concentrate on model-based diagnosis methods that work by trying to restore the consistency of the system description and the observations.

In this section we introduce the standard method for calculating consistency-based diagnosis, presented in [Reiter, 1987]. Although Reiter's framework is based on first-order logic, most of the problems studied in the literature do not make use of full first-order logic and can be easily represented in a propositional language. For the sake of simplicity, we will adapt the definitions given in [Reiter, 1987] to only mention formulas in the propositional language L.

3.1 Basic Definitions

The systems to be diagnosed will be described by a set of propositional formulas. For each component X of the system, we use a propositional variable of the form okX to indicate whether the component is working as it should. If there is no evidence that the system is not working, we can assume that variables of the form okX are true.

Definition 3.1 *A* **system** *is a pair (SD,ASS), where:*

1. *SD, the system description, is a finite set of formulas of L and*
2. *ASS, the set of assumables, is a finite set of propositional variables of the form okX.*

An **observation** is a formula of L. We will sometimes represent a system by (SD,ASS,OBS), where OBS is an observation for the system (SD,ASS).

The need for a diagnosis arises when an abnormal behavior is observed, i.e., when SD∪ASS∪{OBS} is inconsistent. A diagnosis is a minimal set of assumables that must be negated in order to restore consistency.

Definition 3.2 *A* **diagnosis** *for (SD,ASS,OBS) is a minimal set $\Delta \subseteq ASS$ such that:*

$SD \cup \{OBS\} \cup ASS \backslash \Delta \cup \{\neg okX | okX \in \Delta\}$ is consistent.

A diagnosis for a system does not always exist:

Proposition 3.3 *[Reiter, 1987] A diagnosis exists for (SD,ASS,OBS) if and only if $SD \cup \{OBS\}$ is consistent.*

Definition 3.2 can be simplified as follows:

Proposition 3.4 *[Reiter, 1987] The set $\Delta \subseteq ASS$ is a diagnosis for (SD,ASS,OBS) if and only if Δ is a minimal set such that $SD \cup \{OBS\} \cup (ASS \backslash \Delta)$ is consistent.*

3.2 Computing Diagnoses

In this section we will present Reiter's construction for finding diagnoses. Reiter's method for computing diagnosis makes use of the concepts of *conflict sets* and *hitting sets.* A conflict set is a set of assumables that cannot be all true given the observation:

Definition 3.5 *[Reiter, 1987] A* **conflict set** *for (SD,ASS,OBS) is a set $Conf = \{okX_1, okX_2, \ldots, okX_n\} \subseteq ASS$ such that $SD \cup \{OBS\} \cup Conf$ is inconsistent.*

From Proposition 3.4 and Definition 3.5 it follows that $\Delta \subseteq$ASS is a diagnosis for (SD,ASS,OBS) if and only if Δ is a minimal set such that ASS$\backslash\Delta$ is not a conflict set for (SD,ASS,OBS).

A hitting set for a collection of sets is a set that intersects all sets of the collection:

Definition 3.6 *[Reiter, 1987] Let $\mathcal{C}$ be a collection of sets. A* **hitting set** *for $\mathcal{C}$ is a set $H \subseteq \bigcup_{S \in \mathcal{C}} S$ such that for every $S \in \mathcal{C}$, $H \cap S$ is nonempty. A hitting set for $\mathcal{C}$ is minimal if and only if no proper subset of it is a hitting set for $\mathcal{C}$.*

The following theorem presents a constructive approach for finding diagnoses:

Theorem 3.7 *[Reiter, 1987] $\Delta \subseteq ASS$ is a diagnosis for (SD,ASS,OBS) if and only if Δ is a minimal hitting set for the collection of minimal conflict sets for (SD,ASS,OBS).*

> **Example 1:** Consider the circuit in Figure 1. The system description of this circuit is given by (SD,ASS), where:
>
> $$\text{ASS} = \{okX, okY, okZ\}$$
>
> $$\begin{aligned} \text{SD} = \; & \{(A \wedge B) \wedge okX \rightarrow D, \\ & \neg(A \wedge B) \wedge okX \rightarrow \neg D, \\ & C \wedge okY \rightarrow \neg E, \neg C \wedge okY \rightarrow E, \\ & (D \vee E) \wedge okZ \rightarrow F, \\ & \neg(D \vee E) \wedge okZ \rightarrow \neg F\} \end{aligned}$$

Suppose we have OBS=$\neg C \wedge \neg F$. This observation is inconsistent with SD $\cup$ ASS. There is only one minimal conflict set for (SD,ASS,OBS): $\{okY, okZ\}$. There are two minimal hitting sets: $\{okY\}$ and $\{okZ\}$.

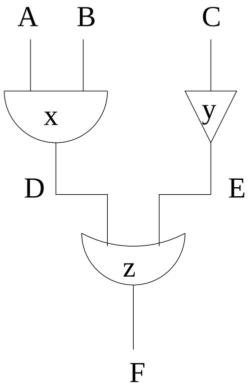

Figure 1: Circuit

4 Diagnosis via Kernel Semi-Revision

The definitions of the last section bear a striking resemblance to those of the operation of kernel semi-revision presented in Section 2.

Recall that kernel operations are based on two concepts: kernels and incision functions. The kernels are the minimal subsets of a belief base implying some sentence, while the incision functions are used to decide which elements of the kernels should be given up. Let (SD,ASS,OBS) be a system. The belief base that we are going to semi-revise corresponds to SD$\cup$ASS and the input sentence is OBS. The conflict sets are the assumables in the inconsistent kernels of SD$\cup$ASS$\cup\{$OBS$\}$. So, if B=SD$\cup$ASS, the conflict sets are given by $\{X\cap\text{ASS} | X \in (B+\text{OBS}) \perp\!\!\!\perp \bot\}$. Incision functions correspond loosely to hitting sets, the minimal hitting sets being the values of minimal incisions that return only assumables. Note that there is a difference in the status of formulas in SD and those in ASS: formulas in ASS represent expectations and are more easily retracted than those in SD (cf. Definition 4.1).

We can model the diagnosis problem as a kernel semi-revision by the observation. Semi-revision can be divided in two steps. First the observation is added to the system description together with the assumables. In case the observation is consistent with the system description together with the assumables, no formula has to be given up. Otherwise, we take the inconsistent kernels and use an incision function to choose which elements of the kernels should be given up.

In the case of diagnosis, we do not wish to give up sentences belonging to the system description or the observation. We prefer to give up the formulas of the form okX, where X is a component of the system. Moreover, we are interested in minimal diagnosis, so the incision should be minimal. For this, we use a slightly different form of incision function. We modify Definition 2.2 so that incisions are minimal and elements of a given set A are prefered over the others:

Definition 4.1 *Given a set A, an A-***minimal incision function** *is any function σ_A from sets of sets of formulas into sets of formulas such that for any set S of sets of formulas:*

1. *$\sigma_A(S) \subseteq \bigcup S$,*
2. *If $\emptyset \neq X \in S$, then $X \cap \sigma_A(S) \neq \emptyset$,*
3. *If for all $X \in S$, $X \cap A \neq \emptyset$, then $\sigma_A(S) \subseteq A$, and*
4. *$\sigma_A(S)$ is a minimal set satisfying 1,2, and 3.*

If we take A to be the set of assumables, we obtain an incision function that prefers to select formulas of the form okX over the others.

We can show that for (SD,ASS,OBS), whenever a diagnosis exists, an ASS-minimal incision function will select only elements of ASS:

Proposition 4.2 *Let (SD,ASS,OBS) be a system with an observation and σ_{ASS} an ASS-minimal incision function. If a diagnosis exists, then*

$\sigma_{ASS}((SD\cup ASS\cup\{OBS\}) \perp\!\!\!\perp \bot) \subseteq ASS$.

Proof: A diagnosis exists if and only if SD is consistent with OBS (Proposition 3.3). Hence, every inconsistent kernel of SD$\cup$ASS$\cup\{$OBS$\}$ must contain an element of ASS. From Definition 4.1, it follows that $\sigma_{ASS}((\text{SD} \cup \text{ASS} \cup \{\text{OBS}\}) \perp\!\!\!\perp \bot) \subseteq$ASS. $\Box$

Lemma 4.3 *The assumables that occur in an inconsistent kernel of the set SD$\cup$ASS$\cup\{OBS\}$ form a conflict set for (SD,ASS,OBS) and all minimal conflict sets can be obtained in this way, i.e.:*

(i) For every $X \in (SD\cup ASS\cup\{OBS\}) \perp\!\!\!\perp \bot$, $X\cap ASS$ is a conflict set, and

(ii) For every minimal conflict set Y, there is some set X such that $X \in (SD\cup ASS\cup\{OBS\}) \perp\!\!\!\perp \bot$ and $X\cap ASS= Y$.

Proof:

(i) Let $X \in$ (SD$\cup$ASS$\cup$\{OBS\}) $\perp\!\!\!\perp \perp$. Then, $X \subseteq$ ($X\cap$ASS)$\cup$SD$\cup$\{OBS\}. Since X is inconsistent, so is ($X\cap$ASS)$\cup$SD$\cup$\{OBS\}, hence $X\cap$ASS is a conflict set.

(ii) Let Y be a minimal conflict set. Then $Y\cup$SD$\cup$\{OBS\} is inconsistent and since $Y \subseteq$ ASS, there is some $X \in$ (SD$\cup$ASS$\cup$\{OBS\}) $\perp\!\!\!\perp \perp$ such that $X\cap$ASS$\subseteq Y$. Suppose by contradiction that there is some formula α such that $\alpha \in Y$ but $\alpha \notin X\cap$ASS. Since $X\cap$ASS is a conflict set for (SD,ASS,OBS), this contradicts the minimality of Y. Hence, $X\cap$ASS$= Y$.
□

Note that not every inconsistent kernel determines a minimal conflict set, since for conflict sets only the elements of ASS matter, i.e., there may be two inconsistent kernels X_1 and X_2 such that $X_1\cap$ASS is a proper subset of $X_2\cap$ASS.

Recall that given an incision function σ, the semi-revision of a set B by a formula α was given by $B?_\sigma\alpha = (B+\alpha) \setminus \sigma((B+\alpha) \perp\!\!\!\perp \perp)$. A diagnosis is given by the elements of ASS that are given up in a kernel semi-revision by the observation.

Proposition 4.4 *Let S=(SD,ASS,OBS) be a system and σ_{ASS} an ASS-minimal incision function.*

(SD $\cup$ASS $\cup$\{OBS\})$\setminus$((SD $\cup$ASS)$?_{\sigma_{ASS}}$OBS) $=$ σ_{ASS}((SD $\cup$ASS $\cup$\{OBS\}) $\perp\!\!\!\perp \perp$) is a diagnosis.

Proof: We have to prove that given a system for which there is a diagnosis and an observation, it holds that:

1. If d is a diagnosis, then there is an ASS-minimal incision function σ_{ASS} such that $d=$ σ_{ASS}((SD $\cup$ASS $\cup$\{OBS\}) $\perp\!\!\!\perp \perp$).
2. If σ_{ASS} is an ASS-minimal incision function, then σ_{ASS}((SD $\cup$ASS $\cup$\{OBS\}) $\perp\!\!\!\perp \perp$) is a diagnosis.

1. Let σ_{ASS}((SD$\cup$ASS$\cup$\{OBS\}) $\perp\!\!\!\perp \perp$)$= d$. We have to show that σ_{ASS} is an ASS-minimal incision function for the relevant domain, i.e., we must show that it satisfies the four conditions of Definition 4.1.

(i) $d \subseteq \bigcup$((SD$\cup$ASS$\cup$\{OBS\}) $\perp\!\!\!\perp \perp$): If d is a diagnosis according to Definition 3.2, then d is a minimal hitting set for the set of all minimal conflicts of (SD,ASS,OBS). From part (ii) of Lemma 4.3 we know that for every minimal conflict set Y, there is $X \in$ (SD$\cup$ASS$\cup$\{OBS\}) $\perp\!\!\!\perp \perp$ such that $X\cap$ASS$= Y$.

(ii) If $\emptyset \neq X \in$ (SD$\cup$ASS$\cup$\{OBS\}) $\perp\!\!\!\perp \perp$, then $X \cap d \neq \emptyset$: From part (i) of Lemma 4.3, we know that $X\cap$ASS is a conflict set. Since d is a hitting set, $X \cap d \neq \emptyset$.

(iii) If $d \not\subseteq$ASS, then $X\cap$ASS$= \emptyset$ for some $X \in$ (SD$\cup$ASS$\cup$\{OBS\}) $\perp\!\!\!\perp \perp$: Since d is a diagnosis, $d \subseteq$ASS and the condition is trivialy satisfied.

(iv) d is a minimal subset satisfying (i),(ii),(iii): Since d is a diagnosis according to Definition 3.2, d is a minimal hitting set.

2. From part (ii) of Lemma 4.3, we have that all minimal conflict sets are elements of the set $\{X\cap$ASS$|X \in$ (SD$\cup$ASS$\cup$\{OBS\}) $\perp\!\!\!\perp \perp\}$. We have to show that an ASS-minimal incision function for the inconsistent kernels determines a minimal hitting set for all minimal conflicts. From Definition 4.1 and Proposition 4.2 it follows that σ_{ASS}((SD$\cup$ASS$\cup$\{OBS\}) $\perp\!\!\!\perp \perp$) is a hitting set for the set of minimal conflicts of (SD,ASS,OBS). That it is also a minimal hitting set follows directly from Definition 4.1. (Since the non-minimal conflict sets contained in $\{X\cap$ASS$|X \in$ (SD$\cup$ASS$\cup$\{OBS\}) $\perp\!\!\!\perp \perp\}$ are supersets of some minimal conflict set and all minimal conflict sets are considered, an ASS-minimal incision function will give the same result as if only minimal conflict sets were considered). □

Going back to the circuit in Figure 1, we see that SD $\cup$ ASS$\cup$ \{OBS\} is inconsistent. This means that SD $\cup$ ASS$\cup$ \{OBS\} has to be consolidated. There is only one inconsistent kernel:

(SD $\cup$ ASS$\cup$\{OBS\}) $\perp\!\!\!\perp \perp$ $= \{\{\neg C \wedge okY \rightarrow E, (D \vee E) \wedge okZ \rightarrow F, okY, okZ, \neg C \wedge \neg F\}\}$

We have two possibilities for ASS-minimal incision functions: $\sigma_1 = \{okY\}$ and $\sigma_2 = \{okZ\}$. This means that either Y or Z are not working well.

5 Reiter's algorithm

The algorithm given in [Reiter, 1987] computes all minimal hitting sets for an arbitrary collection of sets. We will use it later for finding the incision functions used in kernel constructions. We present here the version corrected by [Greiner *et al.*, 1989].

The algorithm generates a directed acyclic graph (DAG) with nodes labeled by sets and arcs labeled by elements of the set. The idea is that for each node labeled by a set S, the arcs leaving from it are labeled by the elements of S. Let $H(n)$ denote the set formed by the labels of the path going from the root to node n. Node n has to be labeled by a set S such that $S \cap H(n) = \emptyset$. If no such set can be found, the node is labeled by @. The idea is that every path finishing at a node labeled by @ is a hitting set, since it intersects all possible labels for the nodes.

The algorithm tries to generate as few new node labels as possible. This is due to the fact that for diagnosis (and for belief revision as well), the collection of sets F which can be used as node labels will be given only implicitly. Calculating one element of F involves a call to a theorem prover to find a conflict set (in the case of diagnosis; a kernel in the case of belief revision) and is therefore a very expensive operation.

The algorithm minimizes the number of calls to the theorem prover by pruning the graph while it is being built. When a new node has to be labeled, the algorithm tries to re-use existing labels first. If a node label S is a superset of another label S', then it can be "closed", it does not have to be considered any longer, since any hitting set for F will be a hitting set for $F \setminus \{S\}$.

Let F be a family of sets.

1. Choose one set to label the root node (level 0).

2. For each node n at level i do:

2.a. If n is labeled by a set S, then for every $s \in S$ create an arc departing from n with label s.

2.b. Set $H(n)$ to be the set of arc labels on the path from the root to node n.

2.c. If there is some node n' such that $H(n') = H(n) \cup \{s\}$, then let the s-arc of n point to n'.

2.d. Else, if there is a node n' labeled by @ such that $H(n') \subset (H(n) \cup \{s\})$ then close the s-arc (i.e., do not compute a label or successors for this node).

2.e. Else, if there is some node n' labeled by S' such that $S' \cap (H(n) \cup \{s\}) = \emptyset$, then let the s-arc of n point to a new node labeled by S'.

2.f. Otherwise, let the s-arc point to a new node m and let m be labeled by the first element S' of F such that $S' \cap H(m) = \emptyset$. If no such set exists, then label m by @.

2.g. If there is some node n' labeled by a set S_1 such that $S' \subset S_1$, then relabel node n' by S' and remove all arcs departing from n' which were labeled by elements of $S' \setminus S_1$.

3. Repeat step 2 for level $i + 1$.

The algorithm expands the graph breadth first. Each level is processed by step 2. Steps 2.c and 2.e re-use nodes or labels if possible. Reiter has proven the following theorem:

Theorem 5.1 *[Reiter, 1987] Let F be a collection of sets and let D be a graph returned by the algorithm above. The set $\{H(n) | n$ is a node of D labeled by @$\}$ is the collection of minimal hitting sets for F.*

The final algorithm for calculating all diagnoses constructs a DAG as above, except that when it is supposed to generate a new node label, it does so by calling the theorem prover with a smaller set. Let TP be a function such that TP(SD,ASS,OBS) returns a conflict set for (SD,ASS,OBS), i.e, a subset S of ASS such that SD∪S∪{OBS} is inconsistent. If no conflict set exists, the function returns @. When one needs to compute a label for a node n, label n by TP(SD,ASS\H(n),OBS).

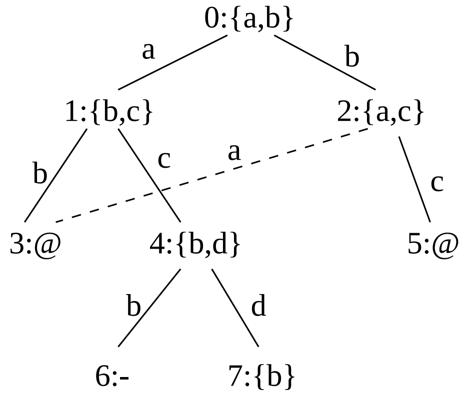

Figure 2: Reiter's algorithm – 1

Consider the following example [Greiner *et al.*, 1989]:

> **Example 2:** Let $F = \{\{a, b\}, \{b, c\}, \{a, c\}, \{b, d\}, \{b\}\}$. Figure 2 shows part of the graph built by the algorithm. The set $\{a, b\}$ is chosen to label node 0 and two arcs are created with labels a and b. Node 1 is labeled by $\{b, c\}$ and node 2 by $\{a, c\}$ and arcs are created leaving from node 1 labeled by b and c and leaving from node 2 labeled by a and c. Node 3 receives the label @, since there is no set $S \in F$ such that $S \cap H(3) = S \cap \{a, b\} = \emptyset$. Node 4 is labeled by $\{b, d\}$. The arc leaving from node 2 and labeled by a points to node 3, since $H(3) = H(2) \cup \{a\}$ (step 2.c of the algorithm). Node 5 is labeled by @, since there is no set $S \in F$ such that $S \cap H(5) = S \cap \{b, c\} = \emptyset$. Node 6 is closed (step 2.d of the algorithm), since nodes 3 and

5 are labeled by @ and $H(3) \subset H(6)$ and $H(5) \subset H(6)$. When node 7 is labeled by $\{b\} \subset \{a, b\}$, the graph is pruned and the root node 0 is relabeled by $\{b\}$ (step 2.g). The resulting graph is shown in Figure 3. The hitting sets are $\{a, b\}$ and $\{b, c\}$.

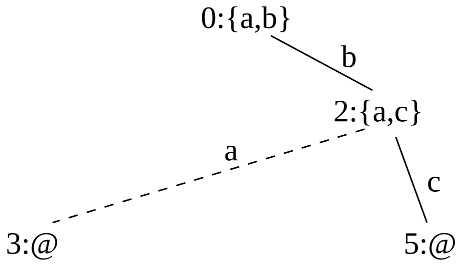

Figure 3: Reiter's algorithm – 2

6 An algorithm for kernel semi-revision

As we have seen, Reiter's algorithm computes all minimal hitting sets for an arbitrary collection of sets. We will use it later for finding the incision functions used in kernel constructions.

In order to apply the algorithm for kernel operations, one needs to adapt very few things. Usually, we will not have access to the whole collection of inconsistent kernels. Using a theorem prover in order to find an inconsistent subset of a belief base does not guarantee that the set returned is a minimal one. Nevertheless, even if the set is not minimal, the algorithm returns the collection of values for the minimal incision functions for all inconsistent subsets of the base. In particular, the returned values are values for incision functions for the inconsistent kernels.

Let TP be a function such that TP(B) returns an inconsistent subset of B. We then build a directed acyclic graph using Reiter's algorithm. Whenever a new label for a node n has to be generated, we call TP(B\H(n)).

That the algorithm does what it is expected to do follows directly from the correctness of Reiter's algorithm.

Example 3: Consider the belief base $B = \{\neg a, \neg b, a \vee b, q, q \rightarrow p, \neg p\}$. There are only two inconsistent kernels, $\{\neg a, \neg b, a \vee b\}$ and $\{q, q \rightarrow p, \neg p\}$. However, the theorem prover may find some superset of these sets. Suppose it finds the collection $\{\{\neg a, \neg b, a \vee b, q\}, \{\neg b, a \vee b, q, q \rightarrow p, \neg p\}, \{\neg a, \neg b, a \vee b, q, \neg p\}, \{\neg a, q, q \rightarrow p, \neg p\}, \{\neg a, \neg b, a \vee b\}, \{q, q \rightarrow p, \neg p\}\}$.

The values for the minimal incision functions are: $\{\neg a, q \rightarrow p\}$, $\{\neg a, q\}$, $\{\neg a, \neg p\}$, $\{\neg b, q \rightarrow p\}$, $\{\neg b, q\}$, $\{\neg b, \neg p\}$, $\{a \vee b, q \rightarrow p\}$, $\{a \vee b, \neg p\}$, and $\{a \vee b, \neg p\}$.

7 Conclusion and Future Work

We have translated a diagnosis problem into a problem of kernel semi-revision where the values of the incision function used must be minimal. It is not difficult to see that kernel operations where the incisions are minimal are equivalent to maxichoice operations. Maxichoice contractions have been shown to have undesirable results when applied to belief sets [Alchourrón *et al.*, 1985]. However, as was argued by Makinson in [Makinson, 1987], they are perfectly acceptable operations when applied to belief bases. This claim can be confirmed by the fact that the diagnosis community, which is not interested in closed theories, has been using maxichoice contraction for finding minimal diagnoses.

We have also shown how Reiter's algorithm for diagnosis can be adapted for implementing belief revision operators. The fact that Reiter's algorithm can be used for belief revision bridges the gap between belief revision theory and implemented systems. Reiter's algorithm is used in several systems and we expect that several computational tools developed for diagnosis systems can be adapted for revision operators.

Reiter's algorithm expands the graph breadth first, generating at the end all possible values for minimal incision functions. If the function TP is substituted by one that finds a minimal inconsistent set, then one can choose to expand depth-first, stopping when one solution was found. If some kind of ordering among the formulas is present (as entrenchment in belief revision or a priori failure probability in diagnosis), this ordering can be used to choose which branch to expand. In this way, it may be possible to obtain partial meet operations by encoding the selection function as the choice of branches to expand.

Acknowledgments

This work is supported by a grant from the Brazilian funding agency CAPES.

References

[Alchourrón and Makinson, 1985] Carlos Alchourrón and David Makinson. On the logic of theory change: Safe contraction. *Studia Logica*, 44:405–422, 1985.

[Alchourrón *et al.*, 1985] Carlos Alchourrón, Peter Gärdenfors, and David Makinson. On the logic of theory change. *Journal of Symbolic Logic*, 50(2):510–530, 1985.

[Fuhrmann, 1991] André Fuhrmann. Theory contraction through base contraction. *Journal of Philosophical Logic*, 20:175–203, 1991.

[Gärdenfors and Rott, 1995] Peter Gärdenfors and Hans Rott. Belief revision. In *Handbook of Logic in Artificial Intelligence and Logic Programming*, volume IV, chapter 4.2. Oxford University Press, 1995.

[Gärdenfors, 1988] Peter Gärdenfors. *Knowledge in Flux - Modeling the Dynamics of Epistemic States.* MIT Press, 1988.

[Greiner *et al.*, 1989] Russell Greiner, Barbara A. Smith, and Ralph W. Wilkerson. A correction to the algorithm in Reiter's theory of diagnosis. *Artificial Intelligence*, 41(1):79–88, 1989.

[Hamscher *et al.*, 1992] Walter Hamscher, Luca Console, and Johan de Kleer, editors. *Readings in Model-Based Diagnosis.* Morgan Kaufmann, 1992.

[Hansson, 1989] Sven Ove Hansson. New operators for theory change. *Theoria*, 55:114–132, 1989.

[Hansson, 1991] Sven Ove Hansson. *Belief Base Dynamics.* PhD thesis, Uppsala University, 1991.

[Hansson, 1992] Sven Ove Hansson. In defense of base contraction. *Synthese*, 91:239–245, 1992.

[Hansson, 1994] Sven Ove Hansson. Kernel contraction. *Journal of Symbolic Logic*, 59:845–859, 1994.

[Hansson, 1997] Sven Ove Hansson. Semi-revision. *Journal of Applied Non-Classical Logic*, 7(1-2):151–175, 1997.

[Hansson, 1999a] Sven Ove Hansson. A survey of non-prioritized belief revision. *Erkenntnis*, 50(2/3):413–427, 1999.

[Hansson, 1999b] Sven Ove Hansson. *A Textbook of Belief Dynamics.* Kluwer Academic Press, 1999.

[Makinson, 1987] David Makinson. On the status of the postulate of recovery in the logic of theory change. *Journal of Philosophical Logic*, 16:383–394, 1987.

[Makinson, 1997] David Makinson. Screened revision. *Theoria*, 63:14–23, 1997.

[Nebel, 1992] Bernhard Nebel. Syntax based approaches to belief revision. In Peter Gärdenfors, editor, *Belief Revision.* Cambridge University Press, 1992.

[Nebel, 1998] Bernhard Nebel. How hard is it to revise a belief base? In D. Dubois and H. Prade, editors, *Handbook of Defeasible Reasoning and Uncertainty Management Systems, Vol. 3: Belief Change*, pages 77–145. Kluwer Academic, 1998.

[Reiter, 1987] Raymond Reiter. A theory of diagnosis from first principles. *Artificial Intelligence*, 32:57–95, 1987. Reprinted in [Hamscher *et al.*, 1992].

[Winslett, 1995] Marianne Winslett. Epistemic aspects of databases. In *Handbook of Logic in Artificial Intelligence and Logic Programming*, volume IV, chapter 4.2, pages 133–174. Oxford University Press, 1995.