A Priori Revision

F. Dupin de Saint-Cyr, B. Duval, S. Loiseau

LERIA, Université d'Angers, 2 bd Lavoisier, 49045 ANGERS Cedex 01 France
{bannay, bd, loiseau}@info.univ-angers.fr

Abstract. The problem of revision is to find which formula ψ can be deduced from a formula φ, which has been added to a Knowledge Base KB. Since φ can bring inconsistency to KB, non-monotonic inference relations which are able to deal with inconsistency have been proposed; note that classical revision takes place after the arrival of φ. The aim of this paper is to propose a priori revision, that is to provide a way to "armor" the KB by suppressing some knowledge and by forbidding to accept some new information in such a way that adding any allowed formula φ to the revised KB will not bring inconsistency.

1 Introduction

A lot of researchers have studied inconsistency handling in knowledge bases (KB for short). The KB is used to describe a system and to deduce new information about it. The difficulty is to reason with an inconsistent KB because the possible deductions become trivial; if we do not want to throw away the whole KB we have to handle inconsistency. A particular problem is the insertion of a new formula in an initially consistent KB; reasoning with the KB after the arrival of this new formula is called *revision* [Alchourrón&al85, Winslett88, Katsuno-Mendelzon91]. So, the problem of revision is to find which formula ψ can be deduced from a formula φ that has been added to the KB. The inference must not be the classical inference since φ can bring inconsistency to the KB. This is why, many researchers have proposed, so called, *non monotonic inference relations* which are able to deal with inconsistency. Those non monotonic inference relations use some *preference relations* that select the most interesting consistent sub-theory(ies) of $\varphi \cup$ KB in which classical deduction can be applied. Note that classical revision takes place *after* the arrival of a new information φ, so this revision can be called *a posteriori revision*.

The aim of this paper is to propose a way to make *a priori revision*. In a priori revision, we want to provide a way to "armor" the KB by suppressing some rules and by forbidding to accept some new information in such a way that adding any allowed formula φ to the revised KB will not bring inconsistency. Consequently, in the revised KB, classical monotonic inference relation will always be usable. In this work, we distinguish between input variables, which can compose a new information, and other variables; we restrict also a new information to be a conjunction of input literals. We propose to examine the initial KB to provide a set of armored KB such that each one will be consistent with any conjunction φ of allowed input literals. A *diagnosis* is composed by a set of formulas that must be removed from the KB and a set of integrity constraints which define *valid new information* for the KB; those integrity constraints provide a way to eliminate some formulas from the set of possible arriving new formulas. Applying a diagnosis to a KB is called *armoring* the KB. One difficulty is that it can exist many such diagnoses. So, we propose to use a penalty preference relation [Dupin&all94] in order to select preferred diagnoses and so armored KB.

This paper is organized as follows. In a first part, we define a priori revision. In the second part, we present a preference relation on diagnoses, based on penalty theory, in order to provide a way to choose a best diagnosis, to obtain a best armored KB. In the last part we propose algorithms to compute the diagnoses and their associated penalty cost.

2 How to armor a knowledge base ?

In the following, we denote by $\mathcal{L}$ a finite propositional language. Elements of $\mathcal{L}$, or *formulas*, are denoted by Greek letters. An *interpretation* in $\mathcal{L}$ is an assignment of a truth value in {T, F} to each formula of $\mathcal{L}$ in accordance with the classical rules of propositional calculus. A literal is an atomic variable p or its negation $\neg$p. An interpretation ω is a *model* of a formula α ($\omega \models \alpha$) iff $\omega(\alpha)$=T. A formula β is called a logical consequence of α ($\alpha \models \beta$) iff each model of α is a model of β. A formula α is said to be consistent iff it has at least one model. Any inconsistent formula can be denoted by $\perp$. A knowledge base KB is a set of logical formulas. Non monotonic inference relation will be denoted by $\mid\!\sim$.

The problem of revision is to decide if, given a knowledge base KB composed by logical formulas and a new information φ, we can deduce ψ, denoted by $\varphi \mid\!\sim_{KB} \psi$. The a posteriori revision selects a set of consistent subsets $KB_{i\ (i=1\ à\ n)}$ of KB such for each subset KB_i, $KBi \cup \varphi \vdash \psi$, which is noted $\varphi \mid\!\sim_{KB} \psi$. The point is to define a preference relation which is able to select the most interesting preferred consistent subsets. In order to discriminate between the consistent subsets of KB, some approaches [Rescher64, Brewka89, Nebel91, Dubois&all92, Benferhat&al93, Lehmann92, Cayrol-Lagasquie95] consist in ranking the KB into priority levels and maximizing the set or the number of formulas satisfied at each level starting from the highest priority level. An important aspect of this kind of approach is that violating however many formulas at a given level is always more acceptable than violating only one formula at a strictly higher level; thus, these approaches are non-compensatory, i.e., levels never interact. An alternative approach, called *penalty* approach, [Pinkas91, Dupin&all94] is to weight the formulas of the KB with positive numbers called penalties. Intuitively, the penalty associated to a formula represents the importance of the formula, the higher it is, the more important is the formula and the more difficult it will be to reject this formula. Inviolable formulas are given an infinite penalty. Contrarily to priorities, penalties are compensatory since they are additive:

the cost associated to a subset of formulas of a KB is the sum of the penalties of the rejected formulas. The subsets having a minimum cost are preferred subsets of KB. Notice that in all these approaches, φ has a maximal priority.

We now present the framework we use in order to make a priori revision. We define a set of *input variables* which is a subset of the variables of $\mathcal{L}$. An *input literal* is an input variable or its negation; we note I this set of input literals. The aim of a priori revision is to compute a revised KB, denoted by D(KB), so that for any valid conjunction of input literals φ which will be added in the future, D(KB) $\cup$ φ will be consistent; hence the classical monotonic inference relation will always be usable with φ. Such a revision of KB is made by defining a diagnosis. A diagnosis is composed of a set of formulas that must be removed from the KB and of a set of integrity constraints which define *valid new information* for the KB. An integrity constraint is a formula $(l_1 \wedge \ldots \wedge l_n \rightarrow \bot)$, where each l_i is an input literal. Such a constraint means that $l_1 \wedge \ldots \wedge l_n$ cannot be added to the KB. To simplify the notations, the constraint formula $(l_1 \wedge \ldots \wedge l_n \rightarrow \bot)$ will be represented by its set of literals $\{ l_1, \ldots, l_n \}$.
We consider the following restrictions: the possible new information is a conjunction of input literals and the knowledge base is a set of *Horn clause* formulas where the positive literal in the clause is not an input literal (we can represent these Horn clauses by implicative formulas where input facts can only occur in the premises). This framework allows us to consider Modus Ponens as the unique inference relation (more formally, with our restrictions, if φ is a conjuction of input literals then $\varphi \cup$ KB infers classically ψ is equivalent to $\varphi \cup$ KB infers ψ by Modus Ponens).

Definition 1 – diagnosis of a knowledge base
Let KB be a knowledge base; I a set of input literals. Let D be a pair $< E_D, r_D >$, where r_D is a subset of KB and E_D is a set of literal sets, $\{\{l_{1,1}; ..., l_{1,n}\}; ...;\{l_{p,1}; ...; l_{p,m}\}\}$ that represents a set of p integrity constraints, called R_{ED}.
D is a *diagnosis* for KB if for every conjunction φ of input literals, consistent with the integrity constraints R_{ED}, $\{\varphi\} \cup$ KB \ r_D is consistent

Example 1 Let us consider the following example: Quakers (Qua) are Pacifists (Pac), Republicans (Rep) are not pacifists, Republicans are American (Am), Americans like Baseball (Bball), and Republicans do not like Baseball. With this knowledge base KB_1, if a new information arrives and states that Nixon is both a Quaker and a Republican, it is possible to deduce that Nixon is both pacifist and not pacifist, a contradiction that we want to avoid.

r1: Qua $\rightarrow$ Pac	r2: Rep $\rightarrow$ ¬Pac	r3: Rep $\rightarrow$ Am
r4: Am $\rightarrow$ Bball	r5: Rep $\rightarrow$ ¬Bball	

If the set of input variables is {Rep, Qua}, then D_0= <{},{r1,r2,r3,r4,r5}>, D_1= <{}, {r1, r3}>, D_2= <{{Rep}}, {}> and D_3= <{{Rep, Qua}}, {r4}> for instance, are possible diagnoses. The computation of diagnoses will be explained in section 3.

An armored KB is a KB on which a diagnosis has been applied.

Definition 2 -armoring a knowledge base
Let D(KB) be the *knowledge base KB armored* by D = < E_D, r_D> ; D(KB) corresponds to KB from which the rules of r_D have been deleted and to which the integrity constraints of R_{ED} are added: D(KB) = $R_{ED} \cup KB \setminus r_D$.

Example 1 If we consider D_9, $D_9(KB_1)$ = { Rep∧Qua→⊥} ∪ {r1, r2, r3, r5}. This means that, with $D_9(KB_1)$, the new information "Nixon is both pacifist and Quaker" represented by Rep∧Qua is forbidden; for any conjunction of input literals j that is not forbidden, j ∪ {r1, r2, r3, r5} is consistent.

3 How to choose the best armoring ?

Definition 3 -a priori revision
A *priori revising* a KB consists in providing a preference relation on the possible diagnoses for a KB.

If there are several diagnoses with the same preference, then either we can define, as in a posteriori revision, that a formula ψ can be inferred from a formula φ if it can be inferred from all the preferred armored KB to which φ is added, or we select any preferred armored KB. A main difficulty is to choose among several possible diagnoses. We propose to use a preference ordering on diagnoses. First, we prefer and so only consider, minimal diagnosis. A minimal diagnosis is a diagnosis that leads to minimal change to the corresponding armored KB. Second, we use a penalty approach that provides criteria to prefer the diagnoses that reject or make useless the less important formulas of KB.

3.1 Minimal change diagnosis

Definition 4 – minimal diagnosis
A diagnosis <E_D, r_D> is *minimal* if there does not exist another diagnosis <E_D', r_D'> verifying: $r_D' \subseteq r_D$, and ($E_D' \subseteq E_D$ or ∀ F' † E_D', ∃ F † E_D such that F ⊆ F').

Examples 1) For the preceding example KB_1, there are 10 minimal diagnoses:
D_1= <{}, {r1, r3}>; D_2= <{}, {r1, r4}>; D_3= <{}, {r1, r5}>; D_4= <{}, {r2, r3}>;
D_5= <{}, {r2, r4}>; D_6= <{}, {r2, r5}>; D_7= <{{Rep}}, {}>;
D_8= <{{Rep, Qua}}, {r3}>; D_9= <{{Rep, Qua}}, {r4}>; D_{10}= <{{Rep, Qua}}, {r5}>
2) Let us suppose that a knowledge base has the three following diagnoses: D1= <{{a,b},{a, c}}, {r1}>, D2=<{{a, b}}, {r1}>, D3=<{{a }}, {r1}>
D2 is minimal, D1 and D3 are not minimal. D1 is not minimal because it is not necessary to forbid the conjunction of the literals a and c to have a diagnosis; D3 is not minimal because D2 shows that it is not necessary to forbid all the interpretations satisfying a, it is sufficient to forbid the interpretations having a and b.
Note that the minimality principle is not interesting for comparing equivalent (in terms of models) diagnoses. For instance, between two diagnoses D1=<{a, b},{a,

$\neg$b}}, {r1}> and D2=<{{a }}, {r1}>, the minimality criterion leads to prefer D1, but in fact the two sets of constraints are equivalent.

We have defined an order relation between diagnoses and the associated minimality criterion. However, this relation only defines a partial order that we propose to refine by using a penalty approach. The penalty approach can rank diagnoses by comparing the weights of the deleted or useless rules.

3.2. Uselessness of a rule in an armored KB

A diagnosis explicitly excludes some rules from the knowledge base. It may also happen that some rules become useless in the revised knowledge base because they can never be fired. A rule cannot be fired if its conditions correspond to an impossible conjunction of input literals or if some of its conditions cannot be proved after the deletion of rules of r_D. If a rule becomes useless after application of a diagnosis, we can consider that the information encapsulated in this rule is, in some way, suppressed from the knowledge base. So, in order to compare armored KB, it is important to know if the rules that are kept in the armored KB are useful.

Definition 5 – useless rule set for a diagnosis D
Let D = < E_D, r_D> be a diagnosis for KB.
A Horn clause r is *useless* for D iff there is no conjunction j of input literals, consistent with R_{ED} , such that the premises of r can be deduced from { j }$\cup$ KB \ r_D .
We call URS(D), useless rule set of D, the set of all useless rules of KB for D.

Example 1 D_1= <{}, {r1, r3}> D_1(KB)={r2, r4, r5}; URS(D_1)={r4}, r4 (Am $\rightarrow$ Bball) is useless because Am is not an input literal, and it cannot be deduced from {r2, r4, r5} with any input base.

Proposition : minimality and uselessness
If D = < E_D, r_D> is a diagnosis for KB, and if r_i in r_D is useless for D, then D is not minimal.

This property means that minimality and uselessness are complementary notions to evaluate diagnoses.

3.3 The penalty approach

For any formula φ_i of the KB, there is an associated penalty $\alpha(\varphi_i)$ that represents a degree of confidence in φ_i, it will be understood as the cost that the user must pay in order to discard the formula φ_i. Let us present the penalty preference on diagnosis. In the basic penalty approach, the philosophy consists in paying $\alpha(\varphi_i)$ when a formula φ_i of the initial knowledge base is discarded, we propose to extend this by taking into account formulas which become useless.

Definition 6 – cost of a diagnosis
Let D = < E_D, r_D> be a diagnosis for KB.
The *cost of the diagnosis* D, called C(D), is the sum of the penalties associated to the rules of KB which are deleted or brought useless by D, so C(D) = $\sum_{j_i \in r_D \cup URS(D)} a(j_i)$

Definition 7 – cost preference
Let KB be a penalty knowledge base. Let D_1 and D_2 be two minimal diagnoses of KB. D_1 is *penalty-preferred* to D_2 iff $C(D_1) \leq C(D_2)$

Example 1 The set of input facts is {Qua, Rep}. We associate a penalty to each rule.
r1: Qua → Pac $\alpha_1 = 5$ r2: Rep → ¬Pac $\alpha_2 = 5$ r3: Rep → Am $\alpha_3 = 100$
r4: Am → Bball $\alpha_4 = 5$ r5: Rep → ¬Bball $\alpha_5 = 7$
The penalty associated to r3 means that this rule is very important.
For each of the minimal diagnoses presented before, we give its cost; the method for computing these costs will be presented in the next section.
D_1= <{}, {r1, r3}> $C(D_1)$=110 (r4 is useless); D_2= <{}, {r1, r4}> $C(D_2)$=10;
D_3= <{}, {r1, r5}> $C(D_3)$=12; D_4= <{}, {r2, r3}> $C(D_4)$=110 (r4 is useless);
D_5= <{}, {r2, r4}> $C(D_5)$=10; D_6= <{}, {r2, r5}> $C(D_6)$=12;
D_7= <{{Rep}}, {}> $C(D_7)$=117 (r2, r3, r4, r7 are useless);
D_8= <{{Rep, Qua}}, {r3}> $C(D_8)$=105 (r4 is useless);
D_9= <{{Rep, Qua}}, {r4}> $C(D_9)$=5; D_{10}= <{{Rep, Qua}}, {r5}> $C(D_{10})$=7.
So, the penalty-preferred minimal diagnosis is D_9, and the associated armored KB is :
Rep∧Qua→⊥ r1: Qua → Pac r2: Rep → ¬Pac
r3: Rep → Am r5: Rep → ¬Bball
It means that if a person is a Quaker then he is a pacifist, and if a person is a Republican then he is non pacifist, american and does not like baseball. But a person can not be both a Quaker and a republican.
Note that taking into account the uselessness of the rules avoids to prefer D_7. The D_7 constraint means that it cannot exist republican, which makes several rules useless.

To apply this approach, a difficulty is to obtain the penalties for all the rules. They can be given by an expert. If no penalty is given, each formula can be associated with a penalty 1, this approach is equivalent to count the number of formulas. An automatic approach can be to associate a penalty to each rule using heuristics. If the KB represents a default behavior of some components, penalties can be proportional to probabilities associated to a faulty component, as [de Kleer-Williams87].

4 Algorithms

Two algorithms are presented. The first one computes the minimal diagnoses of a knowledge base. The second one determines the cost of each diagnosis.

4.1 Diagnosis computation

The computation of diagnoses can be made in two steps using first an ATMS [deKleer86] and second an algorithm that extends [Reiter87]. The first step computes the minimal characterizations of the potential inconsistencies of the KB: such a characterization is a conjunction of input literals and a subset of rules sufficient to infer a contradiction. A conjunction of input literals is also called a fact base and will be represented as a set of literals. Let FB={f1,...,fn} and FB'={f'1,...,f'p} be two fact bases, in the following we denote FB $\models$ FB' the fact that $f1 \wedge ... \wedge fn \models f'1 \wedge ... \wedge f'p$. Notice that [Bouali-Loiseau95] proposes such an algorithm to debug a knowledge base and that [Bezzazi&al98] uses a very similar method for a posteriori revision.

Definition 8 – characterization
Let KB be a knowledge base.
A *characterization* is a pair <FB, rb> where FB is a set of input literals, and rb is a subset of KB, such that $FB \cup rb \models \perp$.
A characterization is *minimal* iff there does not exist another characterization <FB', rb'> such that $rb' \subseteq rb$ and FB $\models$ FB'

ATMS provides a way to compute for each literal a label that defines the necessary and sufficient condition, in terms of assumptions, that provides the deduction of the literal. ATMS provides also a mechanism to ensure that all parts of labels, called environments, are consistent.

Definition 9 – environment and label
An *environment* E is a conjunction of assumptions.
The *label* of a literal L is a disjunction of environments ($E_1 \vee .. E_i ... \vee E_n$) such that :
$\forall E_i$, $E_i \cup KB \not\models \perp$ -the label is consistent with KB (except if $L = \perp$)-, $\forall E_i, E_j$; $E_i \not\models E_j$ -the label is minimal-, $\forall E_i$, $E_i \cup KB \models L$ -the label is sound-, $\forall$ E and $E \cup KB \models L$ then $\exists E_i / E \models E_i$ -the label is complete-.

So given input literals and rules names as assumptions, and the set of rules KB (including for each rule its name as an additional premise), ATMS computes for each literal its label. We denote an environment Ei as composed of Ei_{Rules} the rules figuring in Ei, and Ei_{facts} the facts figuring in Ei. So, the minimal characterizations are composed of the Ei_{facts} part and the Ei_{Rules} part of the environments Ei of $\perp$.

Example 1 Assumptions: {Qua, Rep, r1, r2, r3, r4, r5}; Implications: { $r1 \wedge Qua \rightarrow Pac$, $r2 \wedge Rep \rightarrow \neg Pac$, $r3 \wedge Rep \rightarrow Am$, $r4 \wedge Am \rightarrow Bball$, $r5 \wedge Rep \rightarrow \neg Bball$}

BD_{atms} :

Label(Pac) = (Qua $\wedge$ r1)	Label($\neg$Pac) = (Rep $\wedge$ r2)
Label(Am) = (Rep $\wedge$ r3)	Label(Bball) = (Rep $\wedge$ r3 $\wedge$ r4)
Label($\neg$Bball) = (Rep $\wedge$ r5)	Label(Rep) = (Rep)
Label(Qua) = (Qua)	

Label($\perp$) = (Rep $\wedge$ r3 $\wedge$ r4 $\wedge$ r5)$\vee$ (Qua $\wedge$ Rep $\wedge$ r1 $\wedge$ r2). The environment E2 of label($\perp$) can be noted as $E2_{Rules}$ = {r1, r2} and $E2_{facts}$ = {Qua, Rep}. There exist two minimal characterizations, C1=<{Rep}, {r3, r4, r5}> and C2=<{Qua, Rep},{r1, r2}>.

The diagnoses for a rule base can be computed from the characterizations using the following theorem.

Theorem

Let KB be a rule base, and C = {C1, ..., Cn} the collection of minimal characterizations, $D=<E_D, r_D>$ is a diagnosis w.r.t KB iff $\forall$ $Ci=<E_{Ci}, r_{Ci}>$ of C, either $r_{Ci} \cap r_D \neq \{\}$ or $\exists$ {p1, .., pn} of E_D | E_{Ci} ª {p1,..,pn}

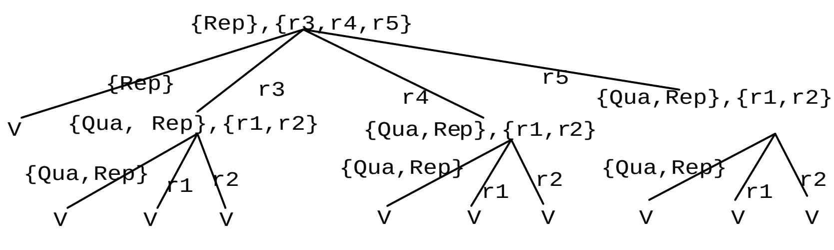
The algorithm which computes the set of minimal diagnoses relative to a rule base from the minimal characterizations is an extension of the algorithm to compute diagnoses [Reiter87]. There are two important differences. First, in the data structure there are two different kinds of data taken into account: the rules and the input literals. A node corresponds to a *characterization*, to each node is associated for each rule it contains an arc labeled by the rule, and for each node is associated an arc labeled with the fact base part of the characterization. Second, the characterizations must be sorted. A diagnosis is obtained by keeping all the labels of arcs from root to a leaf V.

function MinDiag(C): a set of diagnoses

/* C = {C1,...,Cn} is the sorted collection of characterizations; let $C_i = <FB_i, r_i>$, $C_j = <FB_j, r_j>$, $C_i < C_j$ iff FB_j={p'1,...,p'm} |= FB_i = {p1,...,pn}; the function makes a tree whose nodes are labeled by some $C_i=<Fb_i, r_i>$, and the arcs issued of $Ci=<FB_i, r_i>$ are labeled by FB_i or a rule of r_i. *$Hfb(N_i)$ is the set of FB that label the arcs that go from the root to N_i;. $Hr(N_i)$ is the set of rules that label the arcs that are going from the root to N_i */*

```
MinDiag := {}
Label the root of the Tree with the first element of C
For each leaf Nk of Tree labeled by an element Ck=<FBk, rk>
   create a node Nj; create an arc from Nk to Nj labeled by FBk
   For each r of rk
      create node Nj; create an arc from Nk to Nj labeled by rk
   For each node Nj created with an arc from Nk to Nj
      If ∃ Ci=<FBi, ri> of C that verifies Hr(Nj) ∩ ri = {} and
       ∀ {f1,..,fn} of Hfb(Nj), FBi={f'1,...,f'm}|=/= {f1,..,fn}
        Nj := the first Ci verifying the preceding condition
      elseif ∃ Nj' = V, Hr(Nj') ⊆ Hr(Nj), and
       ∀ {f'1,..,f'm} of HFB(Nj'), ∃ {f1,..,fn} of HFB(Nj) /
         {f'1,..,f'm}|= {f1,..,fn}
        close Nj with X
      else close Nj with V, MinDiag:=MinDiag∪{<Hfb(Nj), Hr(Nj)>}
```

Example 1: the schema shows how we find the diagnoses: <{{Rep}}, {}>…



4.2 Diagnosis cost computation

The following algorithm computes the cost of a given diagnosis. It uses the labels (BD_{atms}) computed by ATMS during the diagnosis computation. The cost is composed of the cost of the rules of the diagnosis, plus the cost of the rules that are not useful when the integrity constraints are added and the rules of the diagnosis suppressed.

```
function Cost(D; BDatms) : integer/* Determine the cost of D */
Cost :=0
/* Cost of r of rD*/
For each r of rD
   Cost := Cost + C(r)
/*cost of useless rules*/
For each r of rD
   Suppress all environments of BDatms that contains r
Let RED be the set of the integrity constraints ∧j lij → ⊥
associated to ED
UpDate BDatms by adding the integrity constraints RED
For each r of KB \ rD
   If it does not exist a non contradictory environment in BDatms
   that contains r
      Then Cost := Cost + C(r)
```

Example 1 If we call $Cost(D_1 = <\{\}, \{r1, r3\}>, BD_{atms})$ /* Cost of r of r_D*/ Cost := c(r1) + c(r3) = 5 + 100 /*cost of useless rules*/ BD_{atms} modified: Label(Pac) = $\cancel{(Qua \wedge r1}$); Label(¬Pac) = (Rep ∧ r2); Label(American) = $\cancel{(Rep \wedge r3)}$; Label(Bball) = $\cancel{(Rep \wedge r3 \wedge r4}$); Label(¬Bball) = (Rep ∧ r5); Label(Rep) = (Rep); Label(Qua) = (Qua); Label(⊥) = $\cancel{(Rep \wedge r3 \wedge r4 \wedge r5})\vee$ ($\cancel{Qua \wedge Rep \wedge r1 \wedge r2}$) . It does not exist a non contradictory environment in BD_{atms} modified that contains r4; Cost:=105+C(r4)=110

5 Conclusion

This paper presents a way to armor a knowledge base, by removing some rules and providing some integrity constraints. The adding, in the armored KB, of a new information, consistent with the constraints, does not provide inconsistency, consequently our approach avoids to make non monotonic inference when a new information is added to a KB. So a priori revision is clearly not a AGM revision. Nevertheless it could be interesting to study the links of our approach with contraction [Gärdenfors88].

In previous works [Dupin-Loiseau00], we compared validation versus revision. The validation approach attempts to measure the KB quality so that, if necessary, it can suggest to the expert to improve it. The KB refinement is supported by such a quality measurement. Our new approach for a priori revision extends the notion of diagnosis for validation [Bouali&al97] to diagnosis for revision. The computation of possible diagnoses led us to make restrictions about the syntactical form of the KB and about the new information, these restrictions are directly inspired from the validation field. A point to study is to see if considering any kind of formula as new information is of

any interest for a priori revision. If it is the case, we must study how the algorithms given for a priori revision can be extended in order to deal with any knowledge base in propositional logic.
We can remark that our minimality criterion is purely syntactic and does not recognize that different sets of constraints are equivalent. So further study can examine when it is interesting to propose a reformulation of the set of constraints.

References

[Alchourrón&al85] Alchourrón, Gärdenfors and Makinson. On the logic of theory change: partial meet contraction and revision functions. Symbolic Logic, vol 50, 510-530, 1985.

[Benferhat&al93] Benferhat, Cayrol, Dubois, Lang, Prade. Inconsistency management and prioritized syntax-based entailment. IJCAI, vol 3, 640-645, 1993.

[Benferhat&al95] Benferhat and Smets. Belief functions for logical problems: representing default rules in ε-beliefs logics. In Abstract of the Dagstuhl Seminar, 1993.

[Bezzazi&al98] Bezzazi H., Janot S., Konieczny S., Pino Perez R. Analysing Rational Properties of change operators based on forward chaining. In Transactions and Change in Logic Databases . LNCS Vol. 1472 , pp 317-339.

[Bouali-Loiseau95] Bouali, Loiseau. Rule Base Diagnosis for debugging. EUROVAV, 225-240, 1995.

[Bouali&al97] Bouali, Loiseau, Rousset. Revision of Rule Bases. EUROVAV, 193-204, 1997.

[Brewka89] G. Brewka. Preferred subtheories: an extended logical framework for default reasoning. IJCAI, 1043-1048, 1989.

[Cayrol-Lagasquie95]. Cayrol, Lagasquie-Schiex. Non-monotonic syntax-based entailment: a classification of consequence relations. Lecture notes in AI, 946, p. 107-114, 1995.

[Dubois&al92] Dubois, Lang, Prade. Inconsistency in possibilistic knowledge bases -To live or not live with it. Fuzzy Logic for the Management of Uncertainty, Wiley, 335-351, 1992.

[Dupin&al94] Dupin de Saint-Cyr, Lang and Schiex. Penalty logic and its link with Dempster-Shafer theory. In Proc. of the 10th Uncertainty in AI, p. 204-211, 1994.

[Dupin-Loiseau00] Dupin de Saint-Cyr, Loiseau. Validation et révision. 12ème Congrès Francophone AFRIF-AFIA de RFIA, Paris 1-3 février 2000, vol I, pages 175-183. 2000.

[Gärdenfors88] Gärdenfors P., Knowledge in Flux- Modeling the dynamic of epistemic states. The MIT Press, Cambridge, 1988.

[Katsuno-Mendelzon91] Katsuno, Mendelzon. On the difference between updating a knowledge base and revising it. Principles of Knowledge Representation, p. 387-394, 1991.

[deKleer86] de Kleer, J. An assumption-based truth-maintenance system. Artificial Intelligence, vol.28(2), p.127-224 1986.

[deKleer-Williams87] de Kleer, J. Williams, B.C. Diagnosing Multiple Faults. AI(32), 97-130 1987.

[Lehmann92] D. Lehmann. 1992. Another perspective on default reasoning. Tec. Report.

[Nebel91] B. Nebel. Belief revision and default reasoning: syntax-based approaches. In Proc. of the 2nd KR, p. 417-428. Cambridge, MA, 1991.

[Pinkas91] G. Pinkas. Propositional nonmonotonic reasoning and inconsistency in symmetric neural networks. In Prof. of the 12th IJCAI. p. 525-530. Sydney, Australia, 1991.

[Reiter87] Reiter, R. A theory of diagnosis from first principles. AI, vol 32, p. 57-95, 1987.

[Rescher64] N. Rescher. Hypothetical Reasoning. North-Holland, 1964.

[Winslett88] M. Winslett. Reasoning about action using a possible models approach. AAAI, p.89-93, 1988.