An Algorithm for Knowledge Base Extraction

Alexander Nittka[1]

Abstract. Many approaches have been proposed for reasoning based on conflicting information in general and in particular on *stratified* knowledge bases, i.e. bases in which all pieces of information are assigned a rank. In this paper, we want to reflect on a particular family of *algorithmic* approaches known as *Adjustments*, which have been suggested for extracting a consistent knowledge base from a possibly inconsistent stratified one. We will point out counter-intuitive results provided by these approaches and develop an algorithm we call Refined Disjunctive Maxi-Adjustment which does not have these drawbacks.

1 INTRODUCTION

Reasoning based on conflicting information is one of the main challenges of AI. The problem arises in belief or database merging, belief revision and nonmonotonic reasoning, to name just a few areas. In fact, consistency can never be assumed when modelling an agent interacting with some environment, so inconsistency has to be dealt with. Often the pieces of information available to the agent can be assigned a *reliability*, *priority* or a *rank*. In this special case, the information can be represented by a *stratified knowledge base* $S = (S_1, \ldots, S_n)$, a collection of sets of formulae where each set S_i contains formulae of equal rank, perhaps corresponding to some notion of importance etc. The sets themselves are totally ordered, S_i being more important than S_j for $i < j$. Several approaches to extract a consistent knowledge base from a stratified one have been proposed, [2, 3, 5, 8, 9] to name a few. In this paper we want to reflect on the *algorithmic* presentation of the family of *Adjustments* [3, 8, 9] which construct the consistent knowledge base iteratively, considering one S_i at a time. This form of presentation is especially useful because it makes explicit what causes the decisions in favour or against a formula entering the knowledge base. The most recent and most sophisticated of the approaches is *Disjunctive Maxi-Adjustment* (DMA), which is shown in [3] to be equivalent to the lexicographic system [2, 6].

As an example that this method can lead to counter-intuitive results, consider the following case. Assume that two equally and highly reliable sources provide an agent with convincing evidence, one for b the other for $\neg b$, whereas a less reliable source gives just b. The lexicographic system and DMA – in fact, Maxi-Adjustment [9] as well – tell us that b follows from the corresponding stratified knowledge base $(\{b, \neg b\}, \{b\})$. But why should this be the case? It could be argued that the two equally but more reliable sources disagree and force the agent to be agnostic on the matter and this agnosticism should not be overruled by the information provided by the lesser source. We believe this to be a major problem of these approaches and want to address it in this paper. Consequently, we will present a new algorithm called RDMA – *refined* DMA. For the stratified knowledge base $(\{b, \neg b\})$, DMA decides that there is a tie between b and $\neg b$. With the arrival of information b from a lesser level, this decision is forgotten, allowing b to be inferred. In RDMA, we propose to remember decisions of this kind.

The plan of the paper is as follows. In Section 2 we briefly review the different Adjustments summarized in [3]. Section 3 develops our criticism of these approaches, namely a questionable interpretation of the priorities assigned to formulae belonging to a stratum and the use of definitions inappropriate for the task. Then, in Section 4, we present our solution to the points mentioned before followed by some results concerning RDMA and its relation to the different Adjustments. We conclude and suggest further work in Section 5.

Throughout the paper, we assume a propositional language with the usual connectives. $a, b, \ldots$ denote the propositional variables, $\varphi, \psi, \ldots$ formulae, $C, K, KB, S_i, \ldots$ sets of formulae and $\vdash$ the classical entailment. For sets of formulae K and K', $Cn\,(K)$ denotes the set of conclusions of K, i.e. $Cn(K) = \{\varphi \mid K \vdash \varphi\}$, $|K|$ the cardinality of K, and $K \setminus K'$ the set difference. $\perp$ abbreviates a contradiction. S will usually be a stratified knowledge base $S = (S_1, \ldots, S_n)$.

2 ADJUSTMENTS

For full details on the approaches recalled in this section we refer the reader to [3, 8, 9]. Before presenting the approaches, we want to introduce two important terms they use. Given a set of formulae M, a minimally inconsistent set $C \subseteq M$, i.e. $C \vdash \perp$ and $\forall C' \subset C : C' \not\vdash \perp$, is called a *conflict* in M. The *kernel* of M is the union of all its conflicts.

The basic idea underlying all Adjustments is that the stratified knowledge base is processed stratum by stratum starting with the most important one. The following meta-algorithm illustrates this idea – not all the steps occurring are implemented by every Adjustment variation. Given a stratified knowledge base $S = (S_1, \ldots, S_n)$, the consistent knowledge base KB is calculated as follows:

1. initialize KB
2. for $i \leftarrow 1$ to n do
   (a) identify the consistent part of S_i
   (b) weaken the remaining part of S_i
   (c) update KB
3. return KB

We remark that for all the approaches presented in this paper, the initial knowledge base S is stratified, whereas the resulting one – we will denote it by KB – is not, i.e. Adjustment, (Disjunctive) Maxi-Adjustment and RDMA all calculate a consistent set of formulae. For DMA and RDMA it need not be a subset of formulae contained in S.

[1] Institut für Informatik, Universität Leipzig, Leipzig, Germany
email: nittka@informatik.uni-leipzig.de

2.1 Adjustment

In the most basic approach, which is simply called Adjustment [8] (and which is closely related to [7]), information is added up, starting with the most important, until this would cause an inconsistency. Then the process stops regardless of what is still to come. More formally, if the union of all the strata in $S = (S_1, \ldots, S_n)$ is consistent, then $KB_A = S_1 \cup \cdots \cup S_n$. Otherwise, the union of sets $KB_A = S_1 \cup \cdots \cup S_l$ with l chosen such that $KB_A \not\vdash \bot$ but $KB_A \cup S_{l+1} \vdash \bot$ is taken to be the knowledge base.

Relating this calculation to the meta-algorithm, Adjustment instantiates steps 1, 2a, 2c and 3. It exits the for-loop somewhat uncleanly. If the consistent part of S_i does not coincide with S_i, i is assigned n right away and there is no further update of KB_A.

An argument against this approach is that too much information is discarded as in later sets there may still be information consistent with the base obtained so far. Maxi-Adjustment was proposed to address this shortcoming.

2.2 Maxi-Adjustment

Maxi-adjustment [9] instantiates 1, 2a, 2c, and 3, as well, but improves the unclean exit of the for-loop. It refines Adjustment in that it does not stop when an inconsistency appears. Instead, only the formulae causing the inconsistencies are discarded, the remaining ones are added to the knowledge base. Then the next set is considered.

The calculation starts with $KB_{MA} = \emptyset$ and $i = 1$. At each step we check whether S_i can be consistently included. If yes we do so (KB_{MA} is updated to $KB_{MA} \cup S_i$), if not we add to KB_{MA} only those formulae of S_i which are not involved in any conflict and then proceed in the same way with S_{i+1} until the end of the sequence is reached. This certainly keeps more information than the previous approach, but it can be argued that it still neglects too much of it.

2.3 Disjunctive Maxi-Adjustment

In [3] was proposed an improvement of Maxi-Adjustment. Instead of discarding all the information from S_i involved in a conflict, it could be weakened (via disjunction) until no further conflicts occur.

This modification to Maxi-Adjustment adds a further step (2b) to the algorithm. Before proceeding with S_{i+1}, the formulae of S_i involved in a conflict are considered once more. They themselves cannot be included but weakened versions might. At first all pairwise disjunctions which are not tautologies are tried. If those can be added without causing an inconsistency, this is done. Otherwise, all possible non-tautological disjunctions with three elements are tried, and so on. Other methods of weakening have been proposed, but as the focus of this paper is not on the weakening, we will not go into detail here.

3 PROBLEMS WITH THE ADJUSTMENTS

3.1 Inferring too much

We now give an example and the consistent knowledge bases the three approaches calculate for it. We argue that here the last two approaches possibly allow too much information to enter the knowledge base.

Example 1. *Let $S = (S_1, S_2, S_3)$ where $S_1 = \{b \to a, c \to \neg a\}$, $S_2 = \{b, c\}$ and $S_3 = \{b\}$*

Obviously S_1 is consistent, but trying to add all of S_2 would lead to an inconsistency. So Adjustment accepts S_1 but stops its calculation right afterwards and returns S_1 as result.

MA tries to identify the cause of the inconsistency by calculating the kernel of $S_1 \cup S_2$. In order to do so, all its conflicts are calculated. In this case there is only one, namely $S_1 \cup S_2$ itself. As all elements of S_2 are involved in the conflict, all of them are discarded. The calculation proceeds with S_3, S_1 being the intermediate knowledge base. As $S_1 \cup S_3$ is consistent, this is the result.

DMA weakens the conflicting information before proceeding with S_3. The only possibility to weaken S_2 is to take the disjunction of its two elements. As $b \vee c$ is consistent with S_1, it is included. So the calculation proceeds with S_3, $S_1 \cup \{b \vee c\}$ being the intermediate knowledge base. As before S_3 does not cause an inconsistency, so the final result is $S_1 \cup \{b \vee c\} \cup S_3$. Note that b is element of the knowledge bases calculated by both MA and DMA.

When trying to incorporate S_2 we were forced to leave out both b and c, because we could not decide in favour of one of them. In particular we could not decide in favour of b. In the next step, however, b is added. This means that b wins over c although it has the same or less priority. MA and DMA forget that a negative decision concerning b has already been made. DMA is strongly related to a lexicographic interpretation of the formulae. If there is a tie between two or more on one level the next and less important one may decide, if the comparison using the primary criterion does not provide a winner, we may fall back on a secondary one, and so on. Informally, this strategy is valid if the further criteria add weight to the argument and therefore justify the choice of one object over the other.

We argue that there are cases where this method should not be used, that sometimes such a tie should not be broken. Imagine a support tool used to solve disagreements within a family. The parents have equal priority, the child's opinion is less important. There is to be a nice Saturday dinner with dessert. The mother wants ice cream (a), the father does not ($\neg a$), the child favours ice cream as well. The representation would be $S = (\{a, \neg a\}, \{a\})$ and using DMA, the tool would suggest to have ice cream which is reasonable enough. Now consider the following scenario. The lottery jackpot is astronomical. The father wants to raise a large loan in order to buy as many tickets as possible (a), the mother is totally opposed to that ($\neg a$). The child (for some reason) goes with the father (a). Again $S = (\{a, \neg a\}, \{a\})$ represents the situation, but would it be reasonable to let the vote of the child decide the matter?

We believe that in the second scenario, the vote of the child does not add force to the argument in favour of a, so the matter should be left undecided. If a decision was necessary, it would be more intuitive to consult sources with a higher priority – which are not available in the scenario. The legal system provides a further example where disagreements are generally resolved by referring them to a higher court. In case of contradicting diagnoses concerning a disease one would consult a specialist rather than a general practitioner.

Our intention is to modify the algorithm for DMA to make it applicable to the second scenario. The problem is addressed by carrying along an additional set U. It will collect the formulae which were not added to the knowledge base because they were involved in a conflict. In addition to preventing inconsistency, the algorithm will prevent formulae contained in U from being inferable. This will ensure that no formula for which a negative decision has been made on the basis of a high priority stratum can be added because of a lower stratum. In fact, such a set was already present in the approaches presented so far, but it remained unchanged during the entire calculation, containing a contradiction only.

3.2 Inappropriate definition of conflict

The second point of criticism is the use of what we hold to be an inadequate definition of a conflict. A conflict in a set M is defined as a minimally inconsistent subset of M. This definition presupposes that all elements of M are treated as equal, that any formula can be left out. This is not the case where conflicts are used in (D)MA.

The definition does not reflect the different status of the sets KB (the intermediate knowledge base) and S_i (the set of formulae to be inserted next) with respect to the calculation. In a sense, KB is fixed already – none of its members will leave the set knowledge base during the calculation. Only for elements of S_i is there an option.

Instead of calculating all minimally inconsistent subsets of $KB \cup S_i$, it would be more intuitive to calculate all minimal subsets of S_i inconsistent with KB. The justification is as follows: There is nothing to be done about KB – at this point of time all its elements are accepted to be true. In order to remain consistent we cannot add any set inconsistent with KB. But why only leave out the minimally inconsistent ones? The answer is information economy. We want to keep as much information as possible. Formulae should not be penalized without justification.

Example 2. *Let $S = (S_1, S_2)$ where $S_1 = \{\neg c, b \to a, c \to \neg a\}$ and $S_2 = \{b, c\}$.*

The original definition of a conflict would mark both b and c as causing inconsistencies, because $\{c, \neg c\}$ and $\{b, c, b \to a, c \to \neg a\}$ are conflicts in $S_1 \cup S_2$. Our proposed modification would mark only c. Of course, there is an argument which involves b and leads to a contradiction, but it is based on the assumption that c holds which obviously is not the case. And if this assumption is dropped there is no fault to be found with b, so in the original definition b is penalized because of the unjustified assumption c. For MA, the modification of the definition would make a difference. This can easily be seen in Example 2. In one case b will be left out of the knowledge base, in the other it is included. Whether there are examples where DMA would return different results is subject to future investigations.

In Example 2, the original definition would eliminate b in the first step, but as the weakening $b \vee c$ is consistent with S_1, this disjunction is introduced. Together with $\neg c$, b will be a consequence of the newly found knowledge base. It is possible that this recovery via weakening takes care of the formulae that otherwise might have been penalized, so why bother? There is no reason if we forget which choices we made regarding which formulae to exclude, as the Adjustments do. But as soon as we keep these choices in mind, as we proposed in the last section, we must be careful to choose correctly. If b and c were marked as causing conflicts and therefore not to be inferable, the weakening $b \vee c$ could not be added as then b would be inferable. We want to stress that the counter-intuitive result just sketched is not caused by our proposal not to check for consistency alone, but by the inadequate definition of a conflict.

The question may arise why there should be different results for Example 1 and Example 2. Both seem to express that b and c cannot go together, but both are equally good options. But in fact S_1 in Example 2 says one thing more: c is not an option at all, so it is reasonable to choose b. S_1 in Example 1 does not express a preference, this is why no choice is possible.

4 REFINED DMA

It should be clear that both points of criticism can be dealt with at once or separately – depending on which views are shared. We believe that both should be addressed. We will first give the new definitions for a conflict and the kernel which generalize the original ones Besides extending the term conflict to sets that make certain *marked* formulae inferable, they will reflect the different status of two sets, one that is fixed and one from which formulae can be eliminated. Then we go on to the algorithm.

4.1 (K, U)-conflicts

Definition 4.1. *Let K, M and $U \neq \emptyset$ be sets of formulae.*

- *A set C is a (K, U)-**conflict** if and only if $\exists \psi \in U (C \cup K \vdash \psi) \wedge \forall C' \forall \psi' \in U (C' \subset C \to C' \cup K \not\vdash \psi')$*
- *A set D is (K, U)-**consistent** iff no set $C \subseteq D$ is a (K, U)-conflict.*
- *K is U-consistent if and only if $Cn(K) \cap U = \emptyset$.*
- *If M contains a (K, U)-conflict, then a set $D \subset M$ is **maximally** (K, U)-**consistent** if and only if D is (K, U)-consistent, and every set D' with $D \subset D' \subseteq M$ contains a (K, U)-conflict.*
- *The set $kernel_{(K,U)}(M) = \bigcup_{\substack{C \subseteq M \text{ is a} \\ (K,U)\text{-conflict}}} C$ is the (K, U)-**kernel** of M.*

That is, a (K, U)-conflict $C \subseteq M$ is a minimal set such that some formula contained in U is inferable from $K \cup C$. The kernel collects all sentences of M involved in such conflicts. The U-consistency of K expresses that no element of U can be inferred from K alone, so it generalizes classical consistency in that it refers to arbitrary formulae and not only to a contradiction. Usually we are not interested in whether an arbitrary set is a conflict, but if some set of propositions M contains a conflict; so we will say that C is a conflict in M if C is a conflict and $C \subseteq M$.

Example 3.

- *Let $K = \{d \vee a \to b, b \to c, a \to \neg c\}$ and $U = \{\bot\}$. Then $\{a\}$ is a (K, U)-conflict in $\{a, b, d\}$, whereas $\{b\}$ and $\{a, b\}$ are not.*
- *Let $K = \{b \to c, a \to \neg c\}$ and $U = \{c\}$. Then $\{b\}$ is a (K, U)-conflict in $\{a, b\}$, whereas $\{a\}$ and $\{a, b\}$ are not.*
- *Let $K = \{c, \neg c\}$ and $U = \{a, b\}$. Then $\emptyset$ is the only (K, U)-conflict in $\{a, b, c\}$.*

Proposition 4.2. *M is (K, U)-consistent iff $K \cup M$ is U-consistent.*

Proposition 4.3. *Let $U \neq \emptyset$. If K is U-consistent, then K is consistent.*

Proposition 4.4. *Let K and $U \neq \emptyset$ be arbitrary sets. If C is inconsistent, then C is not (K, U)-consistent.*

Proposition 4.2 tells us that we can safely add a (K, U)-consistent set M to K without affecting the U-consistency. This plays an important part in the algorithm developed in Section 4.2. Propositions 4.3 and 4.4 relate our notion of a conflict to the original definition as well as U-consistency to classical consistency. They show that the definitions we propose are reasonable. Note that for the last two propositions the converse does not hold. A consistent K need not be U-consistent. For example, $\{a\}$ is not $\{a\}$-consistent and although $\{a\}$ is not $(K, U \cup \{a\})$-consistent for arbitrary K and U, $\{a\}$ is consistent.

4.2 RDMA-algorithm

Before presenting to the algorithm, we mark in boldface which modifications to the Adjustment-algorithms presented in Section 2 we propose. Given a stratified knowledge base $S = (S_1, \ldots, S_n)$:

1. initialize KB **and** $\boldsymbol{U}$
2. for $i \leftarrow 1$ to n do
   (a) identify the $(\boldsymbol{KB}, \boldsymbol{U})$-consistent part of S_i
   (b) weaken the remaining part of S_i
   (c) update KB **and** $\boldsymbol{U}$
3. return KB

As proposed in Section 3.1 a set U is used to remember which formulae were excluded from entering the knowledge base. In fact, the update of U is the only new thing but this has a major impact on the remaining essential parts, i.e. 2a and 2b, of the algorithm. U-consistency replaces the classical consistency used in the previous Adjustment-approaches. If we ensure that the knowledge base remains U-consistent at all times, we will know that the resulting knowledge base is consistent and that no unwanted formula is inferable from it.

To complete the algorithm, we need a function implementing the weakening of information. For now, we use the following weakening-function, which is the same as used in DMA: $d_k(C)$ returns the set of all non-tautological disjunctions of size k between different sentences of C if there are any, otherwise the empty set is returned.

Given a stratified knowledge base $S = (S_1, \ldots, S_n)$:

1. $KB \leftarrow \emptyset$
   $U \leftarrow \{\perp\}$
2. **for** $i \leftarrow 1$ **to** n **do**
   (a) $C \leftarrow kernel_{(KB,U)}(S_i)$
   $N \leftarrow S_i \setminus C$
   (b) $k \leftarrow 2$
   while $(k \leq |C| \wedge d_k(C)$ is not $(KB \cup N, U \cup C)$-consistent)
   do $k \leftarrow k+1$
   if $k \leq |C|$ **then** $N \leftarrow N \cup d_k(C)$
   (c) $KB \leftarrow KB \cup N$
   $U \leftarrow U \cup C$
3. return KB

Figure 1. refined disjunctive maxi-adjustment algorithm

Example 4 is to illustrate what the algorithm does. Upper indices indicate the for-loop in which the set was calculated, e.g. C^2 is the kernel calculated during the second loop. This indexing is useful especially for distinguishing the different U and KB.

Example 4. *Let* $S = (S_1, S_2, S_3)$ *with* $S_1 = \{\neg a \vee \neg b, \neg c, \neg d\}$, $S_2 = \{a, b, c, d, e\}$ *and* $S_3 = \{\neg e \vee b\}$

Before the first for-loop is entered we have $KB^0 = \emptyset$ and $U^0 = \{\perp\}$. Now the $(\emptyset, \{\perp\})$-kernel of S_1 must be calculated. As S_1 is consistent, C^1 is empty, $N^1 = S_1$, no weakening is necessary and we enter the next loop with $KB^1 = S_1$ and $U^1 = \{\perp\}$.

S_2 is not (KB^1, U^1)-consistent. The (KB^1, U^1)-conflicts are $\{c\}$, $\{d\}$, and $\{a, b\}$. So all these formulae enter C^2. The only formula not involved in a (KB^1, U^1)-conflict is e which enters N^1.

$d_2(\{a, b, c, d\}) = \{a \vee b, a \vee c, a \vee d, b \vee c, b \vee d, c \vee d\}$ is the first attempt to weakening C^2. Note that $\{c \vee d\}$ is a $(\{\neg a \vee \neg b, \neg c, \neg d, e\}, \{\perp, c, d, a, b\})$-conflict. In fact, it is not even consistent with KB^1. So further weakening is necessary.

Among other disjunctions $d_3(\{a, b, c, d\})$ contains $\{a \vee c \vee d\}$. This is a $(\{\neg a \vee \neg b, \neg c, \neg d, e\}, \{\perp, c, d, a, b\})$-conflict because $\{a \vee c \vee d\} \cup KB^1 \vdash a$ and $a \in U^1$. So we need to consider $d_4(\{a, b, c, d\}) = \{a \vee b \vee c \vee d\}$. This set can be added to to N^2 because it is $(\{\neg a \vee \neg b, \neg c, \neg d, e\}, \{\perp, c, d, a, b\})$-consistent.

Consequently, we have $KB^2 = \{a \vee b \vee c \vee d, \neg a \vee \neg b, \neg c, \neg d, e\}$ and $U^2 = \{\perp, c, d, a, b\}$. As $\{\neg e \vee b\}$ is a (KB^2, U^2)-conflict and cannot be weakened, it is added to U^2. KB^2 remains unchanged, so we have $KB = \{\neg a \vee \neg b, \neg c, \neg d, e, a \vee b \vee c \vee d\}$ and $U = \{\perp, c, d, a, b, \neg e \vee b\}$.

This method of weakening via $d_k(C)$ is open to criticism. It is questionable whether all combinations of elements of the kernel should be considered. It seems more intuitive to weaken only formulae which are tied together by conflicts. Note also that the weakening $a \vee b \vee c \vee d$ in the second for-loop reduces to $a \vee b$ because $\neg c$ and $\neg d$ hold. $a \vee b$ was not accepted directly as in that weakening step there was another conflict. This reminds us of the re-introduction behaviour – giving the right result for the wrong reason – that we criticised in DMA. The reason for this problem is that we did not adjust the weakening-part of the algorithm according to our interpretation. This is beyond the scope of this paper and a subject for future work.

4.3 Properties of RDMA

Ensuring U-consistency throughout the calculation, ensures that the resulting knowledge base will be consistent (Proposition 4.3). This is used to prove Proposition 4.5 which tells us that RDMA does what it is supposed to do. Proposition 4.6 shows that we can ignore multiple occurrences of a formula. We can delete all but the first occurrence of every formula without changing the outcome of the calculation. So we can safely assume that in the stratified knowledge base the intersection of any two sets with different index is empty.

Proposition 4.5. *Let* $S = (S_1, \ldots, S_n)$ *be a stratified knowledge base. Then RDMA calculates a consistent knowledge base.*

Proposition 4.6. *Let the stratified knowledge base* S *be such that* $S = (S_1, \ldots, S_i \cup \{\varphi\}, \ldots, S_{i+j} \cup \{\varphi\}, \ldots, S_n)$, $j \geq 1$. *Then eliminating the second occurrence of* φ *does not change the result of the calculation of the knowledge base* KB. *That is* $S' = (S_1, \ldots, S_i \cup \{\varphi\}, \ldots, S_{i+j}, \ldots, S_n)$ *produces the same* KB.

Note that the property described by Proposition 4.6 does not hold for MA or DMA. This can be seen in Example 1. b appears in S_2 and S_3. If it is eliminated from S_3, it will not be element of the knowledge base calculated, unlike in the original case. Even if we consider only stratified knowledge bases where no formula appears more than once, i.e. $\forall \psi : |\{\, i \,|\psi \in S_i\}| \leq 1$, DMA and our modification do not coincide. The reason is that a formula which has been excluded can still be a consequence of formulae added later on in DMA. This is not possible in RDMA. DMA forgets, RDMA does not.

Example 5. *Consider* $S = (S_1, S_2, S_3)$ *with* $S_1 = \{\neg a \vee \neg b\}$, $S_2 = \{a, b\}$, *and* $S_3 = \{c, c \rightarrow b\}$.

DMA identifies $S_1 \cup S_2$ as a conflict, so S_2 cannot be incorporated into KB but must be weakened. $a \vee b$ is consistent with S_1, so it is added. Then there is no problem with S_3, so the resulting KB is $\{\neg a \vee \neg b, a \vee b, c, c \rightarrow b\}$, from which b can be inferred.

RDMA identifies S_2 as a $(S_1, \{\perp\})$-conflict, so a and b are added to U, but the weakening $a \vee b$ can safely be added to S_1. When considering S_3 it should be clear that it will be possible to infer an element of U, namely b. In fact S_3 is a (KB, U)-conflict. Its only weakening is a tautology, so nothing is added. The knowledge base calculated is $\{\neg a \vee \neg b, a \vee b\}$, from which b cannot be inferred.

$$\begin{tikzcd} & KB_A \arrow[dl] \arrow[d] \arrow[dr] & \\ KB_{MA} & KB_{DMA} & KB_{RDMA} \end{tikzcd}$$

Figure 2. Relation between Adjustments

Figure 2 summarizes the relations between the approaches presented in this paper. An arrow from X to Y is to be read as follows. For an arbitrary stratified knowledge base S we have that $Cn\,(X\,(S)) \subseteq Cn\,(Y\,(S))$. Hence everything that can be inferred from the knowledge base calculated by Adjustment can be inferred from the resulting knowledge bases using the other approaches, but those are mutually incomparable.

For some of the other possible set inclusions it is quite obvious that they cannot hold for all stratified knowledge bases. If weakening of information is allowed, then more information is extracted. It is also obvious that our approach does not generally subsume the conclusions of (D)MA, as it was constructed not to do so.

Most surprising might be that DMA does not always yield all the conclusions of MA. An example is the stratified knowledge base $(\{\neg a \vee \neg b\}, \{a, b\}, \{\neg a, \neg b\})$. The weakening in DMA demands that at least one of the formulae be true. In MA, this is forgotten and the exact opposite is accepted.

But it is also possible for conclusions to be drawn from a knowledge base calculated with our approach that DMA does not allow, although our approach seems much more restrictive. We want to give $S = (\{\neg a \vee \neg b \vee \neg c, \neg a \vee \neg b \vee \neg d\}, \{a, b, c, d\}, \{\neg c\})$ as an example. Note that DMA allows the introduction of c, so allowing $\neg c$ in the end would cause an inconsistency. In RDMA the weakening with pairwise disjunctions is not enough. As a consequence no fault is found when trying to introduce $\neg c$. We do not claim this to be intuitive, far from it. This only shows that RDMA using the current weakening scheme is not *strictly weaker* than DMA. As mentioned before, the weakening needs further investigation and modification.

If all the S_i in the stratified knowledge base are singletons, i.e. $\forall i \leq n : |S_i| = 1$, then MA, DMA and RDMA return identical knowledge bases. For MA and DMA this should be clear, as the only difference is the weakening. As the sets contain only one element, no weakening is possible. For RDMA we only need to make sure that a formula is left out if and only if it causes an inconsistency. As formulae are left out if they allow any element of U to be inferred, we need to show that this is equivalent to causing an inconsistency. This can be done by an easy induction.

We want to remark that the RDMA-algorithm can be seen as the definition for a *removal* operator. This is because U can be initialized to contain more than just a contradiction. The proof of Proposition 4.5 shows that KB, the knowledge base constructed, is U-consistent at all times during the calculation, i.e. no element of U is inferable from KB. The only condition is that KB is initialized to be U-consistent. As KB is empty to begin with, the only requirement is that U cannot contain a tautology, but it is commonly agreed that this a reasonable thing to demand.

The removal operator obtained by allowing U to be initialized differently shows *liberation* behaviour similar to that described in [4]: If the algorithm is run on the same stratified knowledge base S with different initializations for U, e.g. $U = \{\perp\}$ and $U' = \{\psi\}$, it is possible that a formula φ may not be in the knowledge base calculated in the first case where U is used, but be element of the KB when U' is used. The elimination of ψ then led to the liberation of φ.

5 CONCLUSION

In this paper we proposed an new algorithm – RDMA – for extracting a consistent knowledge base from a possibly inconsistent stratified one. This was motivated by counter-intuitive results other approaches yield; they forget negative decisions they made for formulae in strata representing a high priority and consequently may allow them to be introduced based on their reappearance in strata of lower priority. In some cases this reappearance might have a decisive force to break a tie. In others it would be more intuitive if the *additional* information was ignored (note that none of the approaches presented here can deal with scenarios in which these cases are mixed). The intention is to add an approach which can be used in situations where the others fail. Our idea is to remember negative choices by carrying along a second set of formulae that were not allowed to enter the knowledge base and therefore should not be inferable henceforth.

Additionally, we proposed a definition of a conflict that considers the different statuses of the sets involved as well as our notion of remembering choices. It generalizes classical consistency. We presented some results concerning the modified definitions and the RDMA-algorithm.

We did not investigate the nature of the weakening scheme in this paper. As mentioned in connection with Example 4, this is necessary and subject to future work. Further, the relation of RDMA to other schemes for extracting a consistent knowledge base from a stratified one and to argumentation frameworks like that of [1] is of interest. This would shed more light on the reasons of the choices which formulae enter the knowledge base. Another point to be investigated is the computational complexity of the algorithm proposed.

ACKNOWLEDGEMENTS

We are grateful to R. Booth for many insightful discussions and extensive comments on earlier versions of this paper. We want to thank the anonymous referees whose remarks helped to improve the paper.

REFERENCES

[1] L. Amgoud and S. Parsons, ‘An argumentation framework for merging conflicting knowledge bases’, in *Proceedings of the European Conference on Logics in Artificial Intelligence (JELIA’02)*, pp. 27–37, (2002).

[2] S. Benferhat, C. Cayrol, D. Dubois, J. Lang, and H. Prade, ‘Inconsistency management and prioritized syntax-based entailment’, in *Proceedings of the Thirteenth International Joint Conference on Artificial Intelligence (IJCAI’93)*, pp. 640–645, (1993).

[3] S. Benferhat, S. Kaci, D. Le Berre, and M. A. Williams, ‘Weakening conflicting information for iterated revision and knowledge integration’, *Artificial Intelligence*, **153**(1), 339–371, (2004).

[4] R. Booth, S. Chopra, A. Ghose, and T. Meyer, ‘Belief liberation (and retraction)’, in *Proceedings of the Ninth Conference on Theoretical Aspects of Rationality and Knowledge (TARK’03)*, pp. 159–172, (2003).

[5] G. Brewka, ‘Preferred subtheories: An extended logical framework for default reasoning’, in *Proceedings of the Eleventh International Joint Conference on Artificial Intelligence (IJCAI’89)*, pp. 1043–1048, (1989).

[6] D. Lehmann, ‘Another perspective on default reasoning’, *Annals of Mathematics and Artificial Intelligence*, **15**, 61–82, (1995).

[7] J. Pearl, ‘System Z: A natural ordering of defaults with tractable applications to nonmonotonic reasoning’, in *Proc. of the Third Conference on Theoretical Aspects of Reasoning about Knowledge (TARK 1990)*, pp. 121–135, Pacific Grove, CA, (1990).

[8] M. A. Williams, ‘Transmutations of knowledge systems’, in *Proceedings of the Fourth International Conference on Principles of Knowledge Representation and Reasoning (KR’94)*, pp. 619–629, (1994).

[9] M. A. Williams, ‘A practical approach to belief revision: reason-based change’, in *Proceedings of the Fifth International Conference on Principles of Knowledge Representation and Reasoning (KR’96)*, pp. 412–421, (1996).