Computer Science Department Technical Report
Cognitive Systems Laboratory
University of California
Los Angeles, CA 90024-1596

DECIDING CONSISTENCY OF DATABASES CONTAINING DEFEASIBLE AND STRICT INFORMATION

Moises Goldszmidt
Judea Pearl

June 1989
CSD-890034

Submitted for presentation at the Workshop on Defeasible Reasoning with Specificity and Multiple Inheritance, April 7-9, 1989, St. Louis.
Also submitted to 5th Workshop on Uncertainty in AI, Windsor, Ontario.

TECHNICAL REPORT
R-122
April 1989

Deciding Consistency of Databases Containing Defeasible and Strict Information

Moisés Goldszmidt
moises@cs.ucla.edu

Judea Pearl
judea@cs.ucla.edu

Cognitive Systems Lab.
Dept. of Computer Science, UCLA
Los Angeles, CA 90024

Abstract

We propose a formalization of the notion of consistency for a set X of defeasible and strict information, based on a probabilistic interpretation for the sentences in X. This formalization establishes a clear distinction between knowledge bases depicting exceptions and those containing outright contradictions.

A simple and yet powerful theorem is proven giving necessary and sufficient conditions for consistency. This same theorem provides a decision procedure for testing the consistency of X and identifying the inconsistent subset of sentences (in the case that X is inconsistent). Finally, it is shown that if the sentences in X are *Horn* clauses, consistency can be tested in polynomial time.

1 Introduction

There is a sharp difference between exceptions and outright contradictions. Two statements like "typically, penguins do not fly" and "red penguins can fly", can be accepted as a description of a world in which *redness* defines an abnormal type of penguin. However, the statements "typically, birds fly" and "typically, birds do not fly" stand in outright contradiction to each other (unless birds are non existent). Whatever interpretation we give to "typically", it is hard to imagine a world containing birds in which both statements can hold simultaneously. Yet, in spite of this clear distinction, there is no formal treatment of inconsistencies in existing non-monotonic reasoning proposals.

Consider a database Δ containing the following sentences: "all birds fly", "typically, penguins are birds" and "typically, penguins don't fly". A *circumscriptive* theory

([McCarthy, 86]) consisting of the sentences in Δ plus the fact that Tweety is a penguin, will render the conclusion that either Tweety is a flying penguin (and therefore is an exception to the rule "typically, penguins don't fly"), or Tweety is an exception to the rule "typically, penguins are birds" and Tweety does not fly. A formalization of the database in terms of a *default* theory (see [Reiter, 80]) will render similar conclusions for our penguin Tweety. Nevertheless, the above set of rules strike our intuition as being inherently wrong: if all birds fly, there cannot be a nonempty class of objects (penguins) that are "typically birds" and yet "typically, don't fly". We cannot accept this database as merely depicting exceptions between classes of individuals; rather, it would seems that there is no possible state of affairs in which this set of sentences can hold simultaneously[1]. However, if we now change the first sentence of Δ to read "typically, birds fly" instead of "all birds fly", we are willing to cope with the apparent contradiction by considering the set of penguins as an exceptional type of birds. This interpretation will remain satisfactory, even if we change the second rule to read "all penguins are birds". Yet, if we further add to Δ the sentence "typically, birds are penguins" we are faced again with an intuitive *inconsistency*.

This paper deals with the problem of formalizing, detecting and isolating such inconsistencies in knowledge bases containing defeasible and strict information[2]. We will interpret a *defeasible* sentence such as "typically, if ϕ then ψ" written $\phi \rightarrow \psi$, as the conditional probability $P(\psi|\phi) \geq 1 - \varepsilon$, where ε is an infinitesimal quantity[3]. A *strict* sentence such as "if φ it must be the case that σ" written $\varphi \Rightarrow \sigma$, will be interpreted as the conditional probability $P(\sigma|\varphi) = 1$. Our criterion for testing inconsistency translates to that of determining if there is a probability distribution P that satisfies all these conditional probabilities for all $\varepsilon > 0$. To match our intuition that default (strict) sentences do not refer to empty classes, nor are they confirmed by merely falsifying their antecedents, we further require that P be *proper*, i.e., that it does not render any antecedent as totally impossible.

The results, techniques and notation used in this work are based on those presented in [Adams, 75]. In particular, we use the concept of the *quasi-conjunction* and that of a nested sequence of probability assignments.

2 Notation and Preliminary Definitions

We will use ordinary letters from the alphabet (except d, s and x) as propositional variables. Let $\mathcal{F}$ be a factual language built up in the usual way from a finite set of propositional variables and the connectives "$\neg$" and "$\vee$" (the other connectives will be used as syntactic abbreviations), and let the greek letters $\phi, \psi, \varphi, \sigma$ (possibly subscripted) stand for formulas of $\mathcal{F}$.

Let ϕ and ψ be two formulas in $\mathcal{F}$ and let "$\rightarrow$" be a new binary connective, then a

[1] Provided that the set of penguins is nonempty.

[2] The consistency of a system with only defeasible sentences is discussed in [Adams, 75] and [Pearl, 87]

[3] For more on probabilistic semantics for default reasoning the reader is referred to [Pearl, 88].

defeasible sentence is the formula $\phi \rightarrow \psi$ which may be interpreted as "if ϕ then typically ψ". The set of defeasible sentences will be denoted by D. Similarly, given φ, σ in $\mathcal{F}$ and the new binary connective "$\Rightarrow$", a strict sentence is the formula $\varphi \Rightarrow \sigma$ which is to be interpreted as "if φ then it must be the case that σ"[4]. The set of strict sentences will be denoted by S [5]. We will use X to stand for the union of D and S and x, d, s as variables for sentences in X, D and S respectively. Finally, the *material counterpart* of a sentence $\phi \rightarrow \psi$ (or $\phi \Rightarrow \psi$) in X is defined as the formula $\phi \supset \psi$ (where "$\supset$" denotes material implication).

Given a factual language $\mathcal{F}$, a *truth assignment* for $\mathcal{F}$ is a function t, mapping the sentences in $\mathcal{F}$ to the set $\{1,0\}$, (1 for *True* and 0 for *False*), such that t respects the usual boolean connectives. Note that if there are n propositional variables in $\mathcal{F}$, there will be 2^n different truth assignments for $\mathcal{F}$.

A sentence $x \in X$ with antecedent ϕ and consequent ψ will be *verified* by t, if $t(\phi) = t(\psi) = 1$. If $t(\phi) = 1$ but $t(\psi) = 0$, the sentence x will be *falsified* by t. Finally, when $t(\phi) = 0$, x will be considered as neither *verified* nor *falsified*.

Definition 1 (*Probability assignment*). Let P be a probability function on truths assignments, such that $\sum_j P(t_j) = 1$. We define a probability assignment P on a sentence $\phi \rightarrow \psi$ from D as:

$$P(\phi \rightarrow \psi) = \frac{P(t_1)t_1(\phi \wedge \psi) + \cdots + P(t_i)t_i(\phi \wedge \psi)}{P(t_1)t_1(\phi) + \cdots + P(t_i)t_i(\phi)} \tag{1}$$

where $t_1, \ldots, t_i$ are all the possible truth assignments to the propositional variables in $\mathcal{F}$ and $P(t_j)$ is the probability assigned to t_j. We assign probabilities to the sentences in S in exactly the same fashion. P will be considered to be *proper*, if the denominator of Eq (1) is non-zero for every sentence in $D \cup S$.

The definition of probability assignment above, attaches a conditional probability interpretation

$$P(\psi|\phi) = \frac{P(\psi \wedge \phi)}{P(\phi)} \tag{2}$$

to the sentences in X. Eq. (1) states that the probability of a defeasible (strict) sentence $x = \phi \rightarrow \psi$ is equal to the probability of x being verified (i.e. $t_j(\phi \wedge \psi) = 1$), divided by the probability of its being either verified or falsified (i.e. $t_j(\phi) = 1$).

Up to this point the only difference between defeasible sentences and strict sentences was syntactic. They were assigned probabilities in the same fashion and they were verified and falsified under the same truth assignments. Their differences will become clear in the next section, and it rests upon the way they enter the definition of *consistency*.

[4]In the domain of non-monotonic multiple inheritance networks, the interpretation for the defeasible sentence $\phi \rightarrow \psi$ would be "typically ϕ's are ψ's", while the interpretation for the strict sentence $\varphi \Rightarrow \sigma$ would be "all φ's are σ's".

[5]Note that both "$\rightarrow$" and "$\Rightarrow$" can occur only as the main connective.

3 Probabilistic Consistency

Definition 2 (*Probabilistic consistency*) Let D and S be sets of defeasible and strict sentences respectively, constructed from formulas in $\mathcal{F}$. We say that $X = D \cup S$ is *probabilistically consistent* (p-consistent) if, for every $\varepsilon > 0$, there is a proper probability assignment P such that $P(d) \geq 1 - \varepsilon$ for all defeasible sentences d in D, and $P(s) = 1$ for all strict sentences s in S.

Before stating the main result of this paper (theorem 1 below), we need to establish the concepts of *tolerance* and *confirmation* of a sentence x in X.

Definition 3 (*Tolerance*) Let x be a sentence in X with antecedent ϕ and consequent ψ. We will say that x is *tolerated* by the rest of the sentences in X, if there exists a truth assignment t such that the formula $\phi \wedge \psi \wedge X_M$ is satisfied by t where X_M denotes the conjunction of the material counterparts of the sentences in $X - x$.

Definition 4 (*Confirmation*) We will say that a non-empty set of sentences $X = D \cup S$ is *confirmable* when:

1. If D is non-empty, at least one sentence $d \in D$ is tolerated by the rest of the sentences in X.

2. If D is empty, each sentence s in S is tolerated by the rest of the sentences in S.

Theorem 1 Let $X = D \cup S$ be a non-empty set of defeasible and strict sentences constructed from the formulas in $\mathcal{F}$. X is p-consistent if and only if every non-empty subset of X is confirmable.

Proof of the *only if* **part:** We want to show that if there exists a non-empty subset of X which is not confirmable, then X is not p-consistent. The proof is facilitated by introducing the notion of *quasi-conjunction* ([Adams, 75]): Given a set of defaults $D = \{\phi_1 \rightarrow \psi_1, \ldots, \phi_n \rightarrow \psi_n\}$ the *quasi-conjunction* of D is the defeasible sentence,

$$C(D) = [\phi_1 \vee \ldots \vee \phi_n] \rightarrow [(\phi_1 \supset \psi_1) \wedge \ldots \wedge (\phi_n \supset \psi_n)] \tag{3}$$

The quasi-conjunction $C(D)$ bears interesting relations to the set D. In particular, if D is confirmed by some assignment t, $C(D)$ will be verified by t. This is so because the verification of at least one sentence of D by t guarantees that the antecedent of $C(D)$ (i.e. the formula $[\phi_1 \vee \ldots \vee \phi_n]$ in Eq. (3)) is mapped into 1, and the fact that no sentence in D is falsified guarantees that the consequent of $C(D)$ (i.e. the formula $[(\phi_1 \supset \psi_1) \wedge \ldots \wedge (\phi_n \supset \psi_n)]$ in Eq. (3)) is also mapped into 1. Similarly, if at least one sentence of D is falsified, its quasi-conjunction is also falsified. In this case, the consequent of $C(D)$ is mapped into

0 since at least one of the material implication in the conjunction is falsified. Additionally, let $U_p(C(D)) = 1 - P(C(D))$ (the *uncertainty* of $C(D)$) where $P(C(D))$ is the probability assigned to the quasi-conjunction of D according to Eq. (1), then, it is shown in [Adams, 66] that the uncertainty of the quasi-conjunction of D is less or equal to the sum of the uncertainties of each of the sentences in D, i.e. $U_p(C(D)) \leq \sum_i (1 - P(d_i))$ where the sum is taken over all d_i in D.

We are now ready to proceed with the proof. Let $X' = D' \cup S'$ be a subset of X where D' is a subset of D and S' is a subset of S. If X' is not confirmable then one of the following cases must occur:

Case 1.- S' is empty and D' is not confirmable[6]. In this case, the quasi-conjunction for D' is not verifiable; from Eq. (1), we have that $P(C(D')) = 0$ and $U_p(C(D')) = 1$. It follows, by the properties of the quasi-conjunction outlined above that $\sum_i (1 - P(d'_i))$ over all d'_i in D' is at least 1. If the number of sentences in D' is $n > 1$, then,

$$n - \sum_{i=1}^{n} P(d'_i) \geq 1 \quad (4)$$

$$\sum_{i=1}^{n} P(d'_i) \leq n - 1 \quad (5)$$

which implies that at least one sentence in D' has probability smaller than $1 - \frac{1}{n}$ [7]. Hence, it is impossible to have $P(d') \geq 1 - \varepsilon$, for every $\varepsilon > 0$, for every defeasible sentence $d' \in D'$. Thus, X is p-inconsistent.

Case 2.- D' is empty. Proof by contradiction: assume that S' is not confirmable and X' is p-consistent. If X' is p-consistent, there must exist a probability assignment P satisfying definition 2, and a set T of truth assignments such that $P(t_i) > 0$ for all t_i in T. If S' is not confirmable, then either one of the following conditions must be true: there is at least one truth assignment t' in T such that t' falsifies a sentence s' in S', or there is a sentence s'' in S' such that no truth assignment t'' in T verifies s''. The requirements of p-consistency state that for every sentence $\varphi \Rightarrow \sigma$ in S, $P(\varphi \Rightarrow \sigma) = 1$. Thus, from Eq. (1),

$$P(\varphi \Rightarrow \sigma) = \frac{P(t_1)t_1(\varphi \wedge \sigma) + \cdots + P(t_n)t_n(\varphi \wedge \sigma)}{P(t_1)t_1(\varphi) + \cdots + P(t_n)t_n(\varphi)} = 1 \quad (6)$$

which immediately implies that, no sentence $s' \in S'$ can be falsified by any $t \in T$. Hence, the first condition for the unconfirmability of S' cannot occur. On the other hand, if there is no t'' in T that verifies (nor falsifies) a sentence s'' in S', the denominator of $P(s'')$ is 0 (see Eq.(1)), and P is not proper as required. Since by the definition of confirmability these two are the only conditions under which a set of purely strict sentences can be unconfirmable, we conclude that S' cannot be confirmable while X is p-consistent.

Case 3.- Neither D' nor S' are empty and X' is not confirmable. That is, either D' is not confirmable or every t' in T that verifies a sentence in D' falsifies at least one sentence

[6] This case is covered by Theorem 1.1 in [Adams, 75].

[7] If D' consists of only one sentence d', then $P(d') = 0$.

in S'. The first situation will lead us back to case 1 while the second to a contradiction similar to case 2 above. In either case, X is not p-consistent.

Proof of the *if* part: Assume that every non-empty subset of $X = D \cup S$ is confirmable. Then the following two constructions are feasible:

- We can construct a finite "nested decreasing sequence" of non-empty subsets of X, namely $X_1, \ldots, X_m$, $(X = X_1)$, and an associated sequence of truth assignments $t_1, \ldots, t_m$ confirming $X_1, \ldots, X_m$ respectively, with the following characteristics:

  1. X_{i+1} is the proper subset of X_i consisting of all the sentences of D_i not verified by t_i, for $i = 1, \ldots, m-1$.
  2. All sentences in D_m are verified by t_m.

- We can construct a sequence $t_{m+1}, \ldots, t_n$ that will confirm $X_{m+1} = S$. That is, the sequence $t_{m+1}, \ldots, t_n$ will verify every sentence in S without falsifying any[8]. We will associate with $t_{m+1}, \ldots, t_n$ the "nested decreasing sequence" $X_{m+1}, \ldots, X_n$ where X_{i+1} is the proper subset of X_i consisting of all the sentences of S_i not verified by t_i for $i = m+1, \ldots, n$.

We can now assign probabilities to the truth-assignments $t_1, \ldots, t_n$ in the following way:

For $i = 1, \ldots, n-1$

$$p(t_i) = \varepsilon^{i-1}(1-\varepsilon) \tag{7}$$

and

$$p(t_n) = \varepsilon^{n-1} \tag{8}$$

We must show that, in fact, every sentence d in D obtains $P(d) \geq 1 - \varepsilon$ and that every sentence s in S obtains $P(s) = 1$. Since every sentence d is verified in at least one of the member of the sequence $X_1, \ldots, X_n$, using Eq. (1) we have that for $i < n$:

$$P(d_i) \geq \frac{\varepsilon^{i-1}(1-\varepsilon)}{\varepsilon^{i-1}(1-\varepsilon) + \varepsilon^{i}(1-\varepsilon) + \cdots + \varepsilon^{n-1}} = 1 - \varepsilon \tag{9}$$

and $P(d_n) = 1$ if it is only verified by the last truth assignment when S is originally empty. Finally, since no sentence s in S is ever falsified by the sequence of truth assignments $t_1, \ldots, t_n$ and each and every s in S is verified at least once, it follows from Eq (1) and the process by which we assigned probabilities to $t_1, \ldots, t_n$ that indeed $P(s) = 1$ for every $s \in S$.

[8] Note that if S is originally empty, i.e. $X = D$ and $m = n$, then the proof is identical to that of theorem 1.2 in [Adams, 75].

4 Examples

Example 1 On *birds* and *penguins*.

We begin by testing the consistency[9] of the set of sentences presented in the introduction:

1. $b \Rightarrow f$ ("all birds fly").

2. $p \rightarrow b$ ("typically, penguins are birds")

3. $p \rightarrow \neg f$ ("typically, penguins don't fly")

In order for this set of sentences to be consistent, we must find a truth assignment t for the propositional variables b, p and f such that t verifies at least one of the defeasible sentences above (i.e. (2) or (3)) and t does not falsify any of the others. To verify (2), both $t(p)$ and $t(b)$ must be 1. Then, if $t(f) = 1$ sentence (3) is falsified, while if $t(f) = 0$ sentence (1) is falsified. A similar situation arises if we try to verify sentence (3) instead (either (1) of (2) will be falsified). Note that if we change sentence (1) to be defeasible, the truth assignment $t(b) = 1$, $t(f) = 1$ and $t(p) = 0$ will do the job: $b \rightarrow f$ will be verified while neither sentences (2) and (3) will be verified or falsified, hence the set is consistent[10]. If we now add $b \rightarrow p$ ("typically, birds are penguins") the set will again be inconsistent. Any truth assignment verifying one sentence will falsify another.

Example 2 On *quakers* and *republicans*.

Consider the following set of sentences:

1. $w \rightarrow r$ ("typically, bird_watchers are republicans")

2. $w \rightarrow q$ ("typically, bird_watchers are quakers")

3. $q \Rightarrow p$ ("all quakers are pacifists")

4. $r \Rightarrow \neg p$ ("all republicans are non-pacifists")

5. $p \rightarrow c$ ("typically, pacifists are persecuted)

This set of sentences is confirmable by the assignment $t(p) = t(c) = 1$ and $t(w) = t(q) = t(r) = 0$, in which sentence (5) is verified. However, such data base is inconsistent since

[9] The terms *consistency* and *p-consistency* will be used interchangeably.

[10] The reader can verify that any subset can be confirmed.

one of its subsets is not confirmable, namely the set of sentences (1)–(4). Note that theorem 1 not only provides a criteria to decide whether a data base of defeasible and strict information is inconsistent, but also identifies the *offending* set of sentences. We can modify the above set of sentences to be:

1. $w \Rightarrow r$ ("all bird_watchers are republicans")

2. $w \Rightarrow q$ ("all bird_watchers are quakers")

3. $q \rightarrow p$ ("typically, quakers are pacifists")

4. $r \rightarrow \neg p$ ("typically, republicans are non-pacifists")

5. $p \rightarrow c$ ("typically, pacifists are persecuted)

This data base is consistent. There is an important difference between the former case and this one. If all quakers are pacifists and all republicans are non-pacifists, our intuition immediately reacts against the idea of an individual that is both a quaker and a republican. On the other hand, this last set of sentences allows a bird_watcher that is both a quaker and a republican to be either pacifist or non pacifist. Finally, if we make (2) and (4) be the only strict rules, we get a database similar in *structure* to the example depicted by network Γ_6 in [Horty et. al., 88]:

1. $w \rightarrow r$ ("typically bird_watchers are republicans")

2. $w \Rightarrow q$ ("all bird_watchers are quakers")

3. $q \rightarrow p$ ("typically quakers are pacifists")

4. $r \Rightarrow \neg p$ ("all republicans are non-pacifists")

5. $p \rightarrow c$ ("typically pacifists are persecuted)

Not surprisingly, the criterion of theorem 1 renders this database consistent in conformity with the intuition expressed in [Horty et. al., 88].

5 Conclusions

This paper provides a criterion for deciding consistency in data bases containing defeasible and strict information based on a probabilistic interpretation of the sentences in the database. The criterion also identifies the smallest group of sentences that produces the inconsistency. Future work includes a graphical decision criterion for consistency in mixed inheritance networks (extending that of [Pearl, 87]), a formal study of the relation between entailment and p-consistency (see [Adams, 75]), and a comparison to the notion of preferential entailment (see [Lehmann et. al., 88]).

References

[Adams, 75] Adams, E., *The Logic of Conditionals*, chapter II, Dordrecht, Netherlands: D. Reidel.

[Adams, 66] Adams, E., *Probability and The Logic of Conditionals*, in "Aspects of Inductive Logic", ed. J. Hintikka and P. Suppes, Amsterdam: North Holland.

[Dowling et. al., 84] Dowling, W. and J. Gallier, *Linear-Time Algorithms for Testing the Satisfiability of Porpositional Horn Formulae*, Journal of Logic Programming, 3:267–284, 1984.

[Horty et. al., 88] Horty, J. F. and R. H. Thomason, *Mixing Strict and Defeasible Inheritance*, in Proceedings of AAAI-88, St. Paul, Minnesota.

[Lehmann et. al., 88] Lehmann, D. and M. Magidor, *Rational Logics and their Models: A Study in Cumulative Logics*, TR-8816 Dept. of Computer Science, Hebrew University, Jerusalem 91904, Israel.

[McCarthy, 86] McCarthy, J., *Applications of Circumscription to Formalizing Common-sense Knowledge*, Artificial Intelligence, 13:27–39.

[Pearl, 88] Pearl, J., *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*, chapter 10, Morgan Kaufmann Publishers Inc.

[Pearl, 87] Pearl, J., *Deciding Consistency in Inheritance Networks*, Technical Report (R-96) Cognitive Systems Laboratory, UCLA.

[Reiter, 80] Reiter, R., *A Logic for Default Reasoning*, Artificial Intelligence, 13:81–132.

A Appendix: The Complexity of Testing Consistency

Theorem 2 Let $\mathcal{F}$ be a set of propositional formulas, and let $X = D \cup S$ be the a set of defeasible and strict sentences constructed from the formulas in $\mathcal{F}$. The worst case complexity of testing the consistency of X is bounded by $[\mathcal{PS} \times (\frac{|D|^2}{2} + |S|)]$ where $|D|$ and $|S|$ are the number of defeasible and strict sentences respectively, and $\mathcal{PS}$ is the complexity of propositional satisfiability for the material counterpart of the sentences in X.[11]

[11] Although the general satisfiability problem is NP-complete, if the sentences in X are restricted to be Horn clauses then $\mathcal{PS} = O(N)$, where N is the total number of occurrences of literals in X [Dowling et. al., 84]. Thus, for the case of Horn clauses testing consistency will be polynomial.

Proof The following procedure for testing consistency finds a "nested decreasing sequence", (see proof of theorem 1), if one exists; otherwise, it returns failure.

```
PROCEDURE TEST_CONSISTENCY
INPUT: a set X = D ∪ S of defeasible and strict sentences
1.  LET D' := D
2.  WHILE D' is not empty DO
3.          Find a sentence d ∈ D' such that d is
            tolerated by S ∪ D' − d
4.          IF d is found then LET D' := D' − d
            ELSE HALT: the set is INCONSISTENT
ENDWHILE
5.  LET S' := S
6.  WHILE S' is not empty DO
7.          Pick any sentence s ∈ S' and test whether s is
            tolerated by S − s
8.          IF s is tolerated then LET S' := S' − s
9.          ELSE HALT: the set is INCONSISTENT
ENDWHILE
10.  The set is CONSISTENT
END PROCEDURE
```

If the procedure stops at either line (4) or line (9) a non confirmable subset is found, and by theorem 1 the set of sentences is inconsistent. On the other hand, if the procedure reaches line (10), from the proof of the *if* part of theorem 1 we are able to build a proper probability distribution in which all sentences $d \in D$ have $P(d) \geq 1 - \varepsilon$ and all sentences $s \in S$ have $P(s) = 1$; thus the original set X is consistent. It follows that the procedure is correct.

To assess the time complexity, note that the WHILE-loop of line (6) will be executed $|S|$ times in the worst case, and each time we must do at most $\mathcal{PS}$ work to test the satisfiability of $S - s$; thus, its complexity is $|S| \times \mathcal{PS}$. In order to *find* a tolerated sentence $d = \phi \rightarrow \psi$ in D', we must test at most $|D'|$ times, (once for each sentence $d \in D'$), for the satisfiability of the conjunction of $\phi \wedge \psi$ and the material counterparts of the sentences in the set $(S \cup D' - d)$. However, the size of D' is decremented by at least one sentence in each iteration of the WHILE-loop in line (2), therefore the number of times that we test for satisfiability is $|D| + |D| - 1 + |D| - 2 + \ldots + 1$ which is bounded by $\frac{|D|^2}{2}$. The total amount of work in this loop is $O(|D|^2 \times \mathcal{PS})$, and the total time complexity is $O[\mathcal{PS} \times (\frac{|D|^2}{2} + |S|)]$.