COMBINING KNOWLEDGE BASES CONSISTING OF FIRST-ORDER THEORIES

CHITTA BARAL, SARIT KRAUS, JACK MINKER, AND V. S. SUBRAHMANIAN

Department of Computer Science and
University of Maryland Institute for Advanced Computer Studies, University of Maryland, College Park, MD 20742, U.S.

Consider the construction of an expert system by encoding the knowledge of different experts. Suppose the knowledge provided by each expert is encoded into a knowledge base. Then the process of *combining* the knowledge of these different experts is an important and nontrivial problem. We study this problem here when the expert systems are considered to be first-order theories. We present techniques for resolving inconsistencies in such knowledge bases. We also provide algorithms for implementing these techniques.

Key words: knowledge bases, first-order theories.

1. INTRODUCTION

Consider the construction of an expert system by encoding the knowledge of different experts. Suppose that the knowledge provided by each expert is encoded into a knowledge base. Then the process of *combining* the knowledge of these different experts is an important and nontrivial problem.

The problem is important because the user of the expert system so constructed should have access to the knowledge of each of the experts. In particular, he/she should be able to use the knowledge of two different experts to jointly derive a fact that neither of the experts, individually, knew. In other words, one important feature involved in consulting multiple experts is to *pool* their knowledge together and thus obtain knowledge that no individual expert previously had.

The problem is nontrivial because individual experts can, and often do, hold conflicting views on their domain of expertise. Two attorneys involved in a legal defense may well hold conflicting views on the best possible defense strategy, just as two doctors may well differ in their assessment of a patient's malady. In a logic knowledge base, these conflicting opinions manifest themselves in the form of inconsistencies. Classical logic would then indicate that the resulting knowledge base is meaningless—a state of affairs that is clearly inappropriate in this context. Just as the attorneys and doctors would work together to reconcile their views in the interests of the defendant or patient, so should a knowledge base management system reconcile these inconsistencies and allow sensible decisions to be drawn. The key problem here is: how should these inconsistencies be reconciled? This is the problem addressed in this paper.

Baral, Kraus, and Minker (1989) formalize the notion of combining knowledge bases when each knowledge base is a normal Horn logic program (set of rules with only atoms allowed in the head) and assume that the union of the knowledge bases is stratified (no recursion through negation). They assume the presence of world knowledge in the form of integrity constraints, which all the individual knowledge bases satisfy, and the combined knowledge base is required to satisfy. They present methods to obtain a maximally combined knowledge base with respect to the union of knowledge bases that is consistent with respect to the integrity constraints. Since they consider each knowledge base to be a normal Horn logic program, the union of the knowledge bases is always consistent and the combined knowledge base they obtain is maximal with respect to it.

In this paper, we consider each knowledge base as a first-order theory. All knowledge is expressed in the same underlying logical language.[1] The set of integrity constraints is also assumed to be a first-order theory. In this case, the union of the knowledge bases is not necessarily consistent. Because of this, in the absence of integrity constraints, to have the combined knowledge base as the union of the knowledge bases, we need a semantics for inconsistent theories. Many such semantics have been suggested in the past (Blair and Subrahmanian 1988, 1989, da Costa, Subrahmanian, and Vago 1991, Gelfond and Lifschitz, Grant 1974, 1975, 1977, 1978, Kowalski and Sadri, and Subrahmanian 1989). In this paper, we use the cautious approach of Grant and Subrahmanian (1990) to characterize the semantics of inconsistent theories. In the cautious approach, the semantics of an inconsistent theory is the semantics obtained by considering all maximally consistent subsets of the inconsistent theory. A sentence is considered true (false) if it is true (false) in all maximally consistent subsets of the original theory.

In the next section, we will present a scenario that will be used throughout the paper to illustrate the basic intuitions behind our technical development. In section 3 of the paper, we discuss the cautious semantics of inconsistent theories in the presence of integrity constraints. In the subsequent section, we formalize combining a set of theories having the same priority, in the presence of integrity constraints and its relationship with view update approaches (Fagin, Kuper, Ullman, and Vardi 1986 and Fagin, Ullman, and Vardi 1983). We then allow the theories to be prioritized and formalize the notion of combining a set of prioritized theories.

2. A MOTIVATING SCENARIO

Inconsistencies can easily arise when multiple reasoning agents each arrive at a particular view of the world. While these individual views are usually self-consistent, they often tend to conflict with one another. We now present a simple scenario that we will use over and over again to motivate the basic ideas in the paper.

The Scenario: At 1:00 A.M. on January 14, 1990, Don was shot outside the Good Times Bar in Washington. The street was more or less deserted (it being late in the night) except for four people: Don (who got shot), the murderer, and two rather drunk individuals, John and Bill, who were on the street. John is an 85-year-old man who was about 100 yards away from the shooting, while Bill is 30 years old. Bill was about 75 yards away from the shooting. Their stories follow.

John's story:

1. The murderer wore an orange coat.
2. The murderer wore no hat.
3. John knows the murderer got away in a car (as he heard the engine revving up and the car taking off), but he was hiding in a doorway and was too scared to look, and hence cannot tell us anything about the car.

Bill's story:

1. The murderer wore a dark (probably black) coat.
2. The murderer wore no hat.
3. The murderer drove off in a pink Mercedes.

[1]Baral, Kraus, and Minker (1990) present methods to translate between knowledge bases expressed in different logical languages.

If we look at John's story and Bill's story, they are self-consistent. If Bill had not been around, we would probably have accepted John's version of the story (and vice versa). However, their stories conflict with each other (if we make the reasonable assumption that the murderer wore only one coat). This assumption has the status of an integrity constraint: for the purposes of the story, it is a statement that all parties are willing to accept.

Integrity constraints:

1. The murderer wore only one coat at the time of the murder.
2. Based on other evidence, the police present a convincing case that the murderer knew the victim well.
3. Don's close cronies are Jeff, Ed, and Tom.
4. There is no evidence that any of these three individuals had either borrowed or bought a coat recently; so the only coats they could have worn were their own.
5. Jeff and Ed each own a pink Mercedes. Tom doesn't know how to drive.
6. Jeff has an orange coat.
7. Ed has a black coat.
8. There is no possibility of any collusion between Jeff and Ed.

Based on the above story, we are led to suspect Ed or Jeff, but not both. Only one of them was the murderer. If we accept John's story, then Jeff is the murderer. If we accept Bill's story, then Ed is the murderer. We are faced with the following problem: Who did it? There are numerous alternatives.

Alternative 1: In a court of law, the guilt of a person must be established beyond all reasonable doubt. This cannot be established in this case. For example, if Jeff is on trial for the murder of Don, then a reasonable doubt can be cast on his guilt by the defense. A similar situation would occur if Ed were on trial. This situation corresponds to the case where one views each and every possibility in the correctness of the witnesses' statements. We accept a person as guilty if he/she turns out to be guilty in all of these different possible worlds. Thus, according to alternative 1, we can conclude that Jeff or Ed is guilty, but we can neither prove that Jeff is guilty nor can we prove that Ed is guilty.

Alternative 2: We may be led to doubt the correctness of John's statements. After all, he is 85 years old, as compared to Bill's 30 years, and hence, presumably, Bill's eyesight is better. Furthermore, Bill was much closer to the scene of the crime, and hence, one may feel that his evidence is more credible. In deciding who to prosecute (Ed or Jeff), the police may well decide that they can make a more compelling case against Ed based on Bill's evidence. This alternative corresponds to the assignment of a *priority* to Bill's evidence, rather than to John's.

Alternative 3: The third alternative is simply to conclude that the evidence is inconsistent and to use the semantics of classical logic to conclude everything.

These are only three possible scenarios. Each of these represents a reasonable way of reasoning about the body of evidence in front of us. We will illustrate the technical development of the paper by frequent reference to this example.

3. CAUTIOUS SEMANTICS FOR INCONSISTENT THEORIES

We consider a theory to be a set of well-formed formulas. Unless explicitly mentioned otherwise, theories will always be assumed to be *finite*. Several semantics for inconsistent theories have been discussed in Grant and Subrahmanian (1990). One of the approaches to characterize an inconsistent theory is to consider the maximally consistent subsets of the inconsistent theory. A maximally consistent subset of an inconsistent theory T is a theory that is a consistent subset of T, and that becomes inconsistent if any other sentence of T is added to it. Intuitively, each maximally consistent subset of T corresponds to a consistent state of the world that T is trying to characterize. In the presence of a set of worlds, one can either be bold and pick one of them as the "real" state of the world or one can be cautious (some call it skeptical) and consider all of them. Hence, in the cautious characterization of inconsistent theories, the truth value of a sentence L corresponds to the intuition: "Is L true WRT each and every maximally consistent state of affairs?" Maximal consistent subsets correspond to the normal default theories (Reiter 1980, Etherington 1988), and the cautious and bold approaches are similar to the characterizations of default theories based on truth in all its extensions and truth in a particular extension, respectively, due to Etherington (1988).

Consider a theory T whose maximal consistent subsets are $T_1, \ldots, T_n$. In the presence of integrity constraints (world knowledge that every theory has to satisfy), we have to consider only those worlds that are consistent with respect to the integrity constraints. The bold approach to doing this would be to consider only those T_i's, $1 \leq i \leq n$, that satisfy the integrity constraints. Thus, maximal consistent subsets of T that do not satisfy the integrity constraints would be discarded. The cautious approach, on the other hand, would look at each T_i, $1 \leq i \leq n$, and then find maximal subsets of T_i that are consistent with the integrity constraints. An example may clarify this point.

Example 3.1. Suppose that our theory T consists of the following sentences:

1. $p \rightarrow q$
2. $p \rightarrow \neg q$
3. r
4. p

T has three maximally consistent sets:

$$T_1 = \{1,2,3\}$$
$$T_2 = \{1,3,4\}$$
$$T_3 = \{2,3,4\}$$

Suppose that we have as an integrity constraint the single formula $\neg(r \,\&\, p)$. T_1 satisfies IC, but neither T_2 nor T_3 do. Thus, the "bold" approach would discard T_2 and T_3 and select T_1.

The cautious approach would also pick T_1, but it would not discard T_2 and T_3. It would, instead, try to find maximal consistent subsets of T_2 and T_3 that satisfy the integrity constraint. In the case of each of T_2, T_3, this is achieved by discarding either 3 or 4. Hence, the cautious approach would look at the five sets:

T_1
$T_4 = \{1,3\}$ (obtained by discarding 4) from T_2)
$T_5 = \{2,3\}$ (obtained by discarding 4) from T_3)
$T_6 = \{1,4\}$ (obtained by discarding 3) from T_2)
$T_7 = \{2,4\}$ (obtained by discarding 3) from T_3) □

Example 3.2. With respect to the murder example described in the previous section, the cautious semantics would accept the conclusion "X is the murderer" iff X were the murderer irrespective of whether we chose to believe John or Bill. Thus, according to the cautious semantics, there would be no such individual X. However, the cautious semantics would allow us to conclude the sentence "Either Ed is the murderer or Jeff is the murderer" because this follows irrespective of whether we believe John or Bill. □

In the following definitions MAXCONS(P) and MAXCONS(P,IC) are maximal consistent subsets of P and maximal consistent subsets of P with priority to IC. By maximal consistent subsets of P with priority to IC, we mean maximal consistent subsets of $P \cup IC$, which contain all elements of IC.

Definition 3.1 (Maximal Consistency). Let P be a theory, and IC be a set of integrity constraints. A subset $Q \subseteq P$ is said to be *maximally consistent* with priority to IC iff $Q \cup IC$ is consistent, and for every theory Q' such that $Q \subset Q' \subseteq P$, it is the case that $Q' \cup IC$ is inconsistent. MAXCONS(P,IC) is the set of maximally consistent subsets of P with priority to IC. When IC is an empty set, then MAXCONS(P,IC) is called MAXCONS(P) and is the set of maximally consistent subsets of P.

Theorem 3.1 below shows that all theories possess at least one maximal consistent subset with respect to any consistent set of integrity constraints.

Theorem 3.1. Suppose that P is any (possibly infinite) first-order theory and IC is any consistent set of integrity constraints. Then $P \cup IC$ has at least one (possibly infinite) maximally consistent subset P' such that $IC \subseteq P'$. (If P is finite, then its maximal consistent subsets are obviously finite.)

Proof. $P \cup IC$ has at least one consistent subset that is a superset of IC, namely IC itself. Thus, let CONS(P,IC) denote the set $\{X | X \subseteq P \cup IC$ and X is consistent and $IC \subseteq X\}$, and let CONS(P) be defined as CONS(P, $\emptyset$). Thus, CONS(P,IC) $\neq \emptyset$ because $IC \in$ CONS(P,IC). We show below that every ascending chain of elements in CONS(P) has an upper bound in CONS(P). The result then follows from Zorn's lemma.

Suppose that $M_1 \subseteq M_2 \subseteq M_3 \subseteq \ldots$ is an ascending sequence of members of CONS(P), that is, each M_i is a consistent subset of $P \cup IC$ and $IC \subseteq M_i$. Then $M = \cup_{i=1}^{\infty} M_i$ is an upper bound for this ascending sequence. Moreover, M is consistent, $IC \subseteq M$ and $M \subseteq P \cup IC$, that is, $M \in$ CONS(P). The only nonobvious part is the consistency of M.

To see this, suppose that M is not consistent. Then, by the compactness theorem, there is a finite subset $M' \subseteq M$ such that M' is inconsistent. Let $M' = \{\gamma_1, \ldots, \gamma_n\}$ for some integer n. Hence, for each $1 \leq i \leq$ n, there is an integer, denoted α_i such that $\gamma_i \in M_{\alpha(i)}$. Let $\alpha = \max\{\alpha(1), \ldots, \alpha(n)\}$. Then $M' \subseteq M_\alpha$. Hence, as M' is inconsistent, M_α is also inconsistent, thus contradicting our assumption that each M_j, $j \geq 1$, is in CONS(P). □

A weaker version of the above theorem has been established by Grant and Subrahmanian (1990) (see corollary 3.1 below). An important point to note about MAXCONS(P) is that it depends upon the syntactic nature of P. For instance, if we take $P_1 = \{a,b, \neg a\}$ and $P_2 = \{(a \ \& \ b), \neg a\}$, then MAXCONS($P_1$) $= \{\{a,b\}, \{\neg a,b\}\}$, and so it is perfectly reasonable to cautiously conclude b from P_1. On the other hand, MAXCONS(P_2) $= \{\{(a \ \& \ b)\}, \{\neg a\}\}$ from which we cannot cautiously conclude b. Even though the theory $\{(a \ \& \ b)\}$ is logically equivalent to the theory $\{a,b\}$ according to classical logic, this equivalence does not hold when we treat inconsistent theories nonclassically.

The following corollary of Theorem 1 shows that all theories have at least one maximally consistent subset.

Corollary 3.1 (Grant and Subrahmanian 1990). Every theory P has at least one maximally consistent subset.

Proof. Take IC to be the empty set in the proof of theorem 1. □

We now define a notion of entailment based on the cautious approach.

Definition 3.2 ($\vdash_\forall$-Entailment) (Grant and Subrahmanian 1990). Suppose that P is a theory and F is a formula. A notion of entailment, $\vdash_\forall$ based on the cautious approach is defined as follows: $P \vdash_\forall F$ iff $P' \models F$ for every maximal consistent subset $P' \subseteq P$.

We now state some basic properties of $\vdash_\forall$-entailment.

Theorem 3.2 (Grant and Subrahmanian 1990). Suppose that P is a theory and L,L_1,L_2 are ground literals. Then:

1. For all ground literals L, it is not the case that $P \vdash_\forall L$ and $P \vdash_\forall \neg L$.
2. $P \vdash_\forall (L_1 \ \& \ L_2)$ iff $P \vdash_\forall L_1$ and $P \vdash_\forall L_2$.
3. $P \vdash_\forall F$ for all tautologies F of classical logic. □

Example 3.3 Let P be:

$$\begin{gathered} p \vee \neg q \\ \neg p \vee \neg q \\ r \\ q \end{gathered}$$

In this case, MAXCONS(P) consists of three elements:

$\{p \vee \neg q \ ; r \ ; q\}$,
$\{\neg p \vee \neg q \ ; r \ ; q\}$.
$\{p \vee \neg q \ ; \neg p \vee \neg q \ ; r\}$.

In this case, $P \vdash_\forall r$. But $P \nvdash_\forall q$ and $P \nvdash_\forall p$ and $P \nvdash_\forall \neg q$ and $P \nvdash_\forall \neg p$.

□

The appendix contains algorithms that compute MAXCONS(P) and MAXCONS(P,IC). The algorithms assume that the theories have a finite Herbrand base. Function-free databases satisfy this condition. It is possible to determine the Herbrand consistency of function-free databases. A database is Herbrand consistent iff it has a Herbrand model. Note that Herbrand-consistent databases are always consistent in the usual sense of logic, though the converse is not necessarily true. However, it has become a standard practice in the deductive database (or "Datalog") community to consider Herbrand consistency instead of ordinary first-order logical consistency. Throughout the rest of this paper, we will use the word "consistent" to refer to Herbrand consistency.

When an inconsistent theory P consists of n sentences, the algorithm MAXCONS1(P) given in the appendix (Algorithm A.1), constructs its maximal consistent subsets by determining the consistency of each subset of P of cardinality $n - 1$. All such consistent subsets are stored in a set S. For each inconsistent subset, its maximal consistent subsets are added to S. The set of maximal elements of S is MAXCONS(P). The algorithm MAXCONS1(P,IC), which is also in the appendix (Algorithm A.2), extends MAXCONS1(P) by allowing integrity constraints to be present. MAXCONS1(P,IC) tests the consistency of the union of IC with each subset of P and designates some of these subsets as being maximally consistent with respect to IC.

Theorem 3.3 below proves that algorithm MAXCONS1(P) correctly computes the maximal consistent subsets of P.

Theorem 3.3. MAXCONS(P) = MAXCONS1(P)

Proof. [MAXCONS(P) $\subseteq$ MAXCONS1(P)] Suppose $X \in$ MAXCONS(P). Then $X = P - \{C_1, \ldots, C_r\}$ for some integer $r \geq 0$ where $\{C_1, \ldots, C_r\} \subseteq P$. We proceed by induction on r.

Base case: ($r = 0$). In this case, MAXCONS(P) = $\{P\}$ = MAXCONS1(P).

Inductive case: ($r = k + 1$) Consider $P' = P - \{C_{k+1}\}$. Then X is a maximal consistent subset of P', and furthermore, $X = P' - \{C_1 \ldots, C_k\}$. Therefore, by the induction hypothesis, $X \in$ MAXCONS1(P'). As $X \cup \{C_{k+1}\}$ is inconsistent, X is in MAXCONS1(P).

[MAXCONS1(P) $\subseteq$ MAXCONS(P)] Similar. $\square$

The following theorem demonstrates that algorithm MAXCONS1(P,IC) correctly computes the maximal consistent subsets of P with respect to integrity constraints IC.

Theorem 3.4. MAXCONS(P,IC) = MAXCONS1(P,IC)

Proof. Proceeds along exactly the same lines as the proof of Theorem 3. $\square$

Returning to the motivating murder example in Section 2, if we take P to be the union of John's evidence and Bill's evidence, then MAXCONS(P,IC) consists of two theories T_1 and T_2. T_1 contains:

1. All integrity constraints and
2. Sentences 1,2, 3 of John's story and sentences 2, 3 of Bill's story.

Likewise, T_2 contains:

1. All integrity constraints and
2. Sentences 2 and 3 of John's story and sentences, 1,2, 3 of Bill's story.

Thus, using the MAXCONS(P,IC) semantics, we may conclude that the murderer wore no hat (this being true in both T_1 and T_2 above). However, we may not conclude anything about the color of the murderer's coat. We may also conclude that the murderer drove away in a pink Mercedes.

We now discuss a technique for computing MAXCONS(P) in cases when P is a finite set of clauses (a clause is a disjunction of literals). Throughout the rest of this section, we consider only sets of clauses.

Definition 3.3 (Refutations). Suppose that T is a consistent set of clauses and D is a clause such that $T \cup \{D\}$ is inconsistent. A *refutation* of D from T is a sequence $C_1, \ldots, C_n$ such that:

1. C_n is the empty clause $\square$ and
2. Each C_i, $1 \leq i \leq n$ is either in $T \cup \{D\}$ or is a resolvent of two clauses $C, C' \in T \cup \{D\} \cup \{C_1, \ldots, C_{i-1}\}$.

$C_1, \ldots, C_n$ is called a *minimal* refutation of D from T if there is no strict subsequence of $C_1, \ldots, C_n$ which is also a refutation of D from T, that is, there is no sequence $D_1, \ldots, D_m$ such that $\{D_1, \ldots, D_m\} \subset \{C_1, \ldots, C_n\}$ and if $D_i = C_j$ and $D_{i+1} = C_k$, then $j < k$.

Definition 3.4 (Potential Causes). Suppose that T is a consistent set of clauses and D is a clause such that $T \cup \{D\}$ is inconsistent. Let $\mathfrak{R}$ be some minimal refutation of D from T. Then the set $\mathfrak{R} \cap T$ is said to be a *potential cause* of D.

Example 3.4. Suppose that T is the following set of clauses:

$C1$: $a \vee b$
$C2$: $a \vee \neg b$
$C3$: $a \vee c$
$C4$: $a \vee \neg c$

and $D \equiv \neg a$. There are two minimal refutations $\mathfrak{R}_1$ and $\mathfrak{R}_2$ of D from T where:

$$\mathfrak{R}_1 = C1, C2, a, D, \square$$
$$\mathfrak{R}_2 = C3, C4, a, D, \square$$

(Actually, a few more minimal refutations may be obtained by rearranging the occurrences of some of the clauses in $\mathfrak{R}_1, \mathfrak{R}_2$). Thus, the potential causes of D are: $PC_1 = \{C1, C2,\}$ and $PC_2 = \{C3, C4\}$ corresponding to $\mathfrak{R}_1, \mathfrak{R}_2$, respectively. $\square$

In the context of the murder scenario outlined in section 2, there is only one potential cause of the inconsistency, to wit, the following three sentences:

1. John: The murderer wore an orange coat.
2. Bill: The murderer wore a black coat.
3. *IC*: The murderer wore only one coat at the time of the murder.

If T is itself inconsistent, we may talk of refutations of the empty clause, $\square$, from T. This refers simply to different refutations of $\square$ from T. In this case, each minimal refutation of T gives rise to a *potential cause* of the inconsistency of T, to wit, the potential cause of $\square$ WRT, the refutation we are currently considering. Given theorem T, MINS(T) is the set of minimal (WRT inclusion) potential causes of the inconsistency of T.

Algorithm MAXCONS2, which is given in the appendix (Algorithm A.3), presents a different way of computing maximal consistent subsets of P. It performs this computation by identifying potential causes of inconsistencies and deleting some of them. In the remaining lemma and theorems of this section, any unexplained notation will refer to the notation used in the MAXCONS2 algorithm.

Theorem 3.5 below shows that algorithm MAXCONS2 correctly computes maximal consistent subsets of P. We first prove Lemma 3.1 and then use it to prove Theorem 3.5. Lemma 3.1 shows that, when we remove a minimal potential cause of an inconsistency from P, then the remaining theory is consistent.

Lemma 3.1. Suppose that P is a set of clauses and $Z \in$ MINS(P). Then $(P - Z)$ is consistent.

Proof. Suppose $(P - Z)$ is inconsistent. Then there is a minimal refutation of $(P - Z)$. Let $z_1, \ldots, z_m$ be the members of $(P - Z)$ (and hence of P) occurring in this refutation. Therefore, there exists an $1 \leq i \leq n$ such that $S_i = \{z_1, \ldots, z_m\}$. But then, as $Z \in$ MINS, there must exist a $1 \leq j \leq m$ such that $z_j \in Z$. This implies that $z_j \notin (P - Z)$—a contradiction.

Theorem 3.5. MAXCONS(P) = MAXCONS2(P)

Proof. EMAXCONS(P) $\subseteq$ MAXCONS2(P)]. Suppose $X \in$ MAXCONS(P). It suffices to show that $(P - X) \in$ MINS. To do this, we need to show two things.

I. First we show that $(P - X) \cap S_i \neq \emptyset$ for all $1 \leq i \leq n$. Suppose $(P - X) \cap S_i = \emptyset$ for some $1 \leq i \leq n$. Then $S_i \subseteq X$ which contradicts the assumption that X is consistent.

II. Next, we show that $(P - X)$ is a minimal element of S' WRT inclusion. Suppose not. Then there is a $Z \in$ MINS such that $Z \subset (P - X)$. In particular, there exists an $a \in (P - X) - Z$. But then, by lemma 3.1, $(P - Z)$ would be a consistent subset of P. But $X \subset (P - Z)$, thus contradicting the maximality of X.

[MAXCONS2(P) $\subseteq$ MAXCONS(P)] Suppose $X \in$ MAXCONS2(P). Let $S_1, \ldots, S_n$ be all the potential causes of the inconsistency of P. If $n = 0$, then $X = P \in$ MAXCONS2(P).

So assume $n > 0$. Then $X = (P - Y)$ where $Y = \{a_1, \ldots, a_n\}$ and for all $1 \leq i \leq n$, $a_i \in S_i$ and $Y \in$ MINS.

Claim 1: X is consistent.
Proof of Claim 1: Suppose not. Then there exists an $1 \leq i \leq n$ such that $S_i \subseteq X$. In addition, $a_i \in S_i$. But $a_i \notin X$ because $X = (P - Y)$. But this contradicts the statement that $S_i \subseteq X$.

Claim 2: X is maximally consistent.
Proof of Claim 2: Suppose that, instead, there exists a maximal consistent X' such that $X \subset X' \subseteq P$. Let $X = (P - Y)$ and $X' = (P - Y')$. As $X \subset X'$, $Y' \subset Y$. As X' is a maximal consistent subset of P, by the [MAXCONS(P) $\subseteq$ MAXCONS2(P)] part of the proof, $X' \in$ MAXCONS2(P). Thus, $Y' \in$ MINS. But $Y \in$ MINS also. But this is a contradiction because $Y' \subset Y$. □

Algorithm MAXCONS2(P,IC), which is also in the appendix (Algorithm A.4), extends MAXCONS2(P) to the case when integrity constraints are present. The theorem below establishes that MAXCONS2(P,IC) correctly computes the maximal consistent subsets of P with respect to integrity constraints IC.

Theorem 3.6. MAXCONS(P,IC) = MAXCONS2(P,IC)

Proof. Similar to the proof of Theorem 3.5. □

We now prove some results on the complexity of cautious entailment for propositional theories.

Remark. For any n, there exists a propositional theory T of cardinality $2n$ such that the cardinality of MAXCONS(T) is $O(2^n)$.

Proof. For any value of n consider the theory $T_n = \{a_1, \neg a_1, \ldots, a_n, \neg a_n\}$. It can easily be proved that the cardinality of MAXCONS(T_n) is 2^n. □

In order to place complexity results in context, we reiterate the following result due to Plaisted and Vardi.

Theorem 3.7 (Plaisted 1984, Vardi 1982). The Herbrand satisfiability problem for clause form formulas is EXPTIME-complete.

An immediate corollary is the following.

Corollary 3.2. The problem: Given (as input) finite sets T and T' of function-free wffs such that $T \subseteq T'$, determining whether T is a maximal consistent subset of T' is EXPTIME-hard.

Hence, the problem of combining multiple theories seems to be one for which algorithms tend to be exponential in nature.

4. COMBINING GENERAL THEORIES

The problem of combining general theories is formalized as follows. We have a set of consistent theories and a set of integrity constraints.[2] Each theory satisfies the integrity constraints. We would like to combine the given set of theories so that the combined theory is also consistent with respect to the integrity constraints and contains as much consistent information as possible.

The combination of several theories often leads to a multiplicity of resulting theories. Hence, we need to set up a mechanism to work with sets of theories.

Definition 4.1 (Flock). A flock (a term borrowed from Fagin, Kuper, Ullman, and Vardi (1986)) is a set of theories. The flock corresponding to a theory T is a set of subtheories of T.

Usually, when we consider an inconsistent theory T, the flock of interest is the flock of all maximal consistent subsets of T. In order to define combinations of theories, we need to specify the properties that combination functions must satisfy.

Definition 4.2 (Relation between flocks). Let F_1 and F_2 be flocks. $F_1 \leq_1 F_2$ iff $(\forall T \in F_1)(\exists T' \in F_2)$: $T \subseteq T'$.[3]

Definition 4.3 (Consistency). A flock F is said to be consistent with respect to a set of integrity constraints IC iff, for every theory T present in F, $T \cup IC$ is consistent. □

Definition 4.4 (Correctness). A flock F is said to be $\leq_1$-correct with respect to theories $T_1, \cdots, T_k$, if $F \leq_1 \text{MAXCONS}(T_1 \cup \cdots \cup T_k)$. □

Using the properties of consistency and correctness defined above, we now present the concept of combination functions.

Definition 4.5 (Combination of Theories). Let $T_1, \cdots, T_k$ be a set of theories and IC a set of integrity constraints, where each T_1 satisfies IC. A combination function C is a mapping from a set of theories and a set of integrity constraints into a flock satisfying the following three criteria.

1. (Identity) $C(\{T\}, IC) = \{T\}$.
2. (Consistency) $C(\{T_1, \cdots, T_k\}, IC)$ is consistent with respect to IC.

[2]Throughout this paper, we assume that, whenever we are combining theories $T_1, \ldots, T_n$ and WRT, the set IC of integrity constraints, then each T_i, $1 \leq i \leq n$, is consistent. Note, however, that different T_i, T_j may be mutually inconsistent for $1 \leq i, j \leq n$, $i \neq j$.

[3]Many other orderings between flocks can also be defined. One such ordering is defined as: $F_1 \leq_2 F_2$ iff $(\forall T \in F_1)(\exists T' \in F_2) : Cn(T) \subseteq Cn(T')$, where $Cn(T)$ is the set of classical logic consequences of T. We do not explore this in this paper.

3. (Correctness) $C(\{T_1, \cdots, T_k\}, IC)$ is $\leq_1$-correct with respect to the theories $T_1 \cdots, T_k$.

Another useful property is associativity, which is defined as follows.

$$C(\{T_1, \cdots, T_j, C(\{T_{j+1}, \cdots, T_k\}, IC)\}, IC)$$
$$=_{mm} C(\{C(\{T_1, \cdots, T_j\}, IC), T_{j+1}, \cdots, T_k\}, IC$$
$$=_{mm} C(\{T_1, \cdots, T_k\}, IC); \text{ where } P =_{mm} Q \text{ means } Cn(P) = Cn(Q). \quad \square$$

We now define three combination functions. The first two are skeptical in nature. The first combination function takes the union of the theories and the integrity constraints and looks at the maximal consistent subsets of this union with priority to the integrity constraints. More formally,

Definition 4.6. $\text{Comb}_1(\{T_1, \cdots, T_k\}, IC) \stackrel{\text{def}}{=} \text{MAXCONS}(T_1 \cup \cdots \cup T_k, IC)$.

We now show that Comb_1 is a valid combination function.

Theorem 4.1. Comb_1 is a combination function, that is, it satisfies the identity, consistency, and correctness criteria. Furthermore, Comb_1 is associative. □

Note that the combination function Comb_1 takes the union of theories $T_1, \ldots, T_k$ and then finds the maximal subsets that are consistent with the integrity constraints, IC. We now apply Comb_1 to the murder example.

Example 4.1 (Murder Example on Comb_1). Returning to the murder example. In this case, Comb_1 would take the two witness accounts (John and Bill) as two theories. Let us consider John's three statements to constitute theory T_1 and Bill's three statements to constitute theory T_2. In addition the set of eight integrity constraints is denoted IC. Comb_1 would try to identify maximal subsets of $T_1 \cup T_2$ that are consistent with IC. There would be two maximal consistent subsets, and the flock $\text{Comb}_1(\{T_1,T_2\},IC)$ would consist of these two sets. The difference between the two is that one of these maximal sets contains the statement "The murderer wore an orange coat" (this set would tend to implicate Jeff as the murderer), while the other contains the statement "The murderer wore a dark coat" (this set would implicate Ed). □

The second combination function takes the union of the theories and finds its maximal consistent subsets. It then looks at all the theories in $\text{MAXCONS}(Y_i,IC)$ for all $Y_i \in \text{MAXCONS}(T_i, \cup \cdots \cup T_k)$.

Definition 4.7. $\text{Comb}_2(\{T_1, \cdots, T_k\},IC) \stackrel{\text{def}}{=}$ maximal elements of S, where $S = \{X : X \in \text{MAXCONS}(Y_i,IC)$ where $Y_i \in \text{MAXCONS}(T_1 \cup \cdots \cup T_k)\}$.

The following example illustrates the two approaches.

Example 4.2 Let the union of the theories T be:

a
b
c
$\neg a \vee \neg b$

and the set of integrity constraints be $IC = \{\neg a \vee \neg c\}$

$$\text{MAXCONS}(T,IC) = \{\{a, b\}, \{a, \neg a \vee \neg b\}, \{b, c, \neg a \vee \neg b\}\}$$
$$\text{MAXCONS}(T) = \{T_1 = \{a,b,c\}, T_2 = \{a, c, \neg a \vee \neg b\}, T_3 = \{b, c, \neg a \vee \neg b\}\}$$
$$\text{MAXCONS}(T_1,IC) = \{\{a, b\}, \{c, b\}\}$$
$$\text{MAXCONS}(T_2,IC) = \{\{a, \neg a \vee \neg b\}, \{c, \neg a \vee \neg b\}\}$$
$$\text{MAXCONS}(T_3,IC) = \{\{b, c, \neg a \vee \neg b\}\}$$

Hence, in this example, MAXCONS(T, IC) = maximal members of the set $\cup_{T_i \in \text{MAXCONS}(T)} \text{MAXCONS}(T_i,IC)$.

$a \vee b$ is true in all models of members of MAXCONS(T,IC) and hence it is true with respect to the combination of these theories. □

We now use the murder example to illustrate the behavior of Comb_2.

Example 4.3 (Murder Example on Comb_2). Consider how Comb_2 behaves WRT the murder scenario of section 2. Comb_2 would first construct MAXCONS(P) where P is the union of John's story and Bill's story. Note that P is perfectly consistent and hence MAXCONS(P) = $\{P\}$. (The fact that the murderer wore only one coat at the time of the murder is necessary for the inconsistency to arise.) Comb_2 now computes the maximal elements of $\{X \mid X \in \text{MAXCONS}(Y_i,IC)$ where $Y_i \in \text{MAXCONS}(P)\}$. In this case, this leads to exactly the same results as Comb_1. □

That these two seemingly different combination functions lead to the same results is not an accident, as shown in the following theorem.

Theorem 4.2 MAXCONS(T,IC) = maximal members of the set S where $S = \{X : X \in \text{MAXCONS}(T_i,IC)$ where $T_i \in \text{MAXCONS}(T)\}$

Proof.

$\subseteq$

Let $X \in \text{MAXCONS}(T,IC)$. Then there exists a Y in MAXCONS(T) such that: (1) $X \cap T \subseteq Y$ and (2) $X \in \text{MAXCONS}(Y,IC)$.

Consider such a Y. Suppose $X \notin \text{MAXCONS}(Y,IC)$. Since $X \cup IC$ is consistent, this means that there is an X' such that $X \subset X'$, and such that $X' \in \text{MAXCONS}(Y,IC)$, but then our assumption that $X \in \text{MAXCONS}(T,IC)$ is contradicted. Hence, $X \in$ MAX-CONS(Y,IC).

This proves that $X \in S$. Since, $X \in \text{MAXCONS}(T,IC)$, it has to be a maximal member of S.

$\supseteq$

Suppose X is a maximal element of S. Then there is a $T_i \in \text{MAXCONS}(T)$ such that $X \in \text{MAXCONS}(T_i,IC)$ and no Y such that $X \subset Y$ and $Y \in \text{MAXCONS}(T_j,IC)$ for some $T_j \in \text{MAXCONS}(T)$. This implies that X is consistent with IC.

Suppose $X \notin \text{MAXCONS}(T,IC)$. Then there is an α such that $X \subset \alpha$ and $\alpha \in \text{MAXCONS}(T,IC)$. By the first part of the theorem, this means that $\alpha \in$ MAXCONS (T_j,IC), for some $T_j \in \text{MAXCONS}(T)$. This violates our initial assumption about X, which says that no such T_j exists: a contradiction. Hence, $X \in \text{MAXCONS}(T,IC)$. □

Corollary 4.1. Comb_2 is an associative combination function. □

The third function uses a bold approach. Here we consider any maximal consistent subset of the union of the given theories that satisfies the integrity constraints. It is defined as follows.

Definition 4.8. $\mathrm{Comb}_3(\{T_1, \cdots, T_k\}, IC \stackrel{\mathrm{def}}{=} \{X : X \in \mathrm{MAXCONS}(T_1 \cup \cdots \cup T_k)$ and $X \cup IC$ is consistent.$\}$

Considering Example 4.2, we have $\mathrm{MAXCONS}(T) = \{T_1 = \{a,b,c\}, T_2 = \{a,c,\neg a \vee \neg b\}, T_3 = \{b,c,\neg a \vee \neg b\}\}$.

Of the three theories in $\mathrm{MAXCONS}(T)$, only T_3 is consistent with respect to $IC = \{\neg a, \vee \neg c\}$. Hence, the third approach considers only T_3.

Unlike Comb_1 and Comb_2, Comb_3 is not associative.

Theorem 4.3. Comb_3 is a nonassociative combination funciton. □

The proof that Comb_3 is a combination function is a simple verification of the three conditions that a combination function must satisfy. The example below demonstrates the nonassociative nature of Comb_3.

Example 4.4. Consider the theories T_1,T_2,T_3 and the set IC of integrity constraints shown below:

$$T_1 = \{(a \vee b), c\}$$
$$T_2 = \{\neg a\}$$
$$T_3 = \{a, \neg c\}$$
$$IC = \{\neg b\}$$

In this case, combining T_1 and T_2 first and then combining the result with T_3 yields a flock consisting of one set only:
$C(\{C(\{T_1,T_2\},IC),T_3\},IC) = \{\{a,\neg c\}\}$.

However, combining T_1 with the combination of T_2 and T_3 first and then combining the result with T_1 yields a flock containing three sets:

$$C(\{T_1,C(\{T_2,T_3\},IC)\},IC) = \{\{a \vee b),a,c\}, \{(a \vee b),a,\neg c\}, \{c,\neg a\}\}.$$

This shows that combining the theories T_1,T_2,T_3 in different orders leads to different results. □

One may wonder what the relationship between combining theories using Comb_1 and using Comb_3 is. The theorem below shows that Comb_3 may be viewed as a "bolder" combination strategy than Comb_1.

Theorem 4.4. Let $\{T_1, \ldots, T_k\}$ be a flock and IC a set of integrity constraints. Then:

$$\mathrm{Comb}_3(\{T_1, \ldots, T_k\},IC) \subseteq \mathrm{Comb}_1(\{T_1, \ldots, T_k\},IC).$$

Proof. Suppose $T \in \mathrm{Comb}_3(\{T_1, \ldots, T_k\},IC)$. Then $T \in \mathrm{MAXCONS}(T_1 \cup \cdots \cup T_k)$ and $T \cup IC$ is consistent. Consequently, $T \in \mathrm{MAXCONS}(T_1 \cup \cdots \cup T_k,IC)$. Hence, $T \in \mathrm{Comb}_1\{T_1, \ldots, T_k\},IC)$. □

The distinction between Comb_1 and Comb_3 may be illustrated further by the following example:

Example 4.5. Consider the theories T_1, T_2, and the set of integrity constraints IC shown below:

$$T_1 = \{(a \vee b), (\neg a \vee c)\}$$
$$T_2 = \{\neg b\}$$
$$IC = \{\neg c\}$$

In this case, $\text{Comb}_1(\{T_1, T_2\}, IC)$ consists of the following three theories:

$$\{\neg c, (a \vee b), (\neg a \vee c)\}$$
$$\{\neg c, (a \vee b), \neg b\}$$
$$\{\neg c, (\neg a \vee c), \neg b\}$$

On the other hand, $\text{Comb}_3(\{T_1, T_2\}, IC)$ is the empty flock. □

Comb_3, when applied to the murder example, yields a different result than Comb_1 and Comb_2.

Example 4.6 (Murder Example WRT Comb_3). Using Comb_3, the murder example would result in the empty flock. This is because the union of John's story and Bill's story is consistent; however, the union is no longer consistent with IC and hence is rejected by Comb_3. □

The salient difference between Comb_3 and Comb_1 (and hence also Comb_2) is that the former:

is nonassociative and
is bolder than Comb_1 as it picks a subset of the set of theories picked by Comb_1.

The nonassociativity of Comb_3 restricts the applicability of Comb_3 to situations where we want to combine theories according to a particular ordering. Such a situation may arise, for instance, if all theories are ranked, and we always combine them, starting from the lowest rank upwards. On the other hand, it appears to be easier, in practice, to compute Comb_3 than it is to compute Comp_1. In the case of Comb_3, we simply check if the maximally consistent subsets of $T_1 \cup \cdots \cup T_k$ are consistent with the integrity constraints. Let $Y_1, \ldots, Y_m$ be these maximal consistent subsets. Those Y_i's, $1 \leq i \leq m$ that are not consistent with IC are discarded. The remainder are retained. In the case of Comb_2, however, we would first (as before) find the maximal consistent subsets $Y_1, \ldots, Y_m$ of $T_i \cup \cdots \cup T_k$. We would then try to find maximal subsets of each of the Y_i's that are consistent with IC. This is a more time consuming and laborious operation than simply checking if the Y_i's are consistent with IC. Thus, we would recommend that, in time-critical situations, Comb_3 could be first computed to give a quick "forecast" of the results. If time remains to do a more detailed analysis, then Comb_1 could be used.

4.1 A More Practical Approach

In the previous combination functions, we searched for all the maximally consistent subsets of the inconsistent theories. This is an extremely time-consuming process since it requires consistency checks in each step of the algorithm (Algorithm A.2). In this section, we would like to present a more practical algorithm.

In addition, we are motivated by the following argument. We are combining several theories. Each of them is consistent, and the inconsistency arose from the union of the theories. We do not know which of the theories being combined contains erroneous information. We would like to search for a combination function that will include as much information as possible from the original theories. By "as much," we mean as many clauses as possible, that is, theories that are maximal WRT cardinality rather than with respect to inclusion.

The algorithm MAXCONS3 given in the appendix (Algorithm A.5) serves both purposes. Given theories $T_1, \ldots, T_k$ and a set IC of integrity constraints, it computes the largest possible subset (in terms of cardinality) of $(T_1 \cup \cdots \cup T_k)$ that are consistent with IC.

Definition 4.9. $\text{Comb}_4(\{T_1, \cdots, T_k\}, IC) \stackrel{\text{def}}{=} \text{MAXCONS3}(T_1 \cup \cdots \cup T_k, IC)$.

The intuitive motivation behind Comb_4 is that "every time we discard something, we are (implicitly) saying that the discarded object constitutes a mistake made by the expert. As this information is obtained from experts, it seems appropriate to assume that they don't make too many mistakes; hence we want to pick those maximal consistent subsets where the number of mistakes is minimized." Comb_4 thus seems to be suitable in situations where it is appropriate to make this assumption. The theorem below shows that Comb_4 is bolder than Comb_1.

Theorem 4.5. $\text{Comb}_4(\{T_1, \cdots, T_k\}, IC)$ = maximal sets WRT cardinality of $\text{Comb}_1(\{T_1, \cdots, T_k\}, IC)$.

Comb_4 coincides with Comb_1 on the murder example. The following example illustrates the difference between this approach and the MAXCONS(P) approach.

Example 4.7. Consider the theory P below:

d
$\neg d$
$d \rightarrow e$
$d \rightarrow \neg e$

MAXCONS(P) contains exactly one element, $\{\neg d \rightarrow e; d \rightarrow \neg e; \neg d\}$. Note, however, that, in addition to this set, MAXCONS(P) contains $\{d \rightarrow e; d\}$, $\{d \rightarrow \neg e; d\}$, $\{d, d \rightarrow e\}$. □

The following result is easy to establish.

Corollary 4.2. Let P be any first-order theory. MAXCONS3(P) $\subseteq$ MAXCON(P). □

We note that the inferences of MAXCONS3 are less cautious and more optimistic than those of MAXCONS: that is, everything that is inferred from MAXCONS is also inferred from MAXCONS3, but there is more information that is inferred from MAXCONS3 than from MAXCONS. A straightforward verification demonstrates that Comb_4 is a combination function.

$$\begin{array}{ccc} \mathrm{Comb}_3 & \subseteq & \mathrm{Comb}_1 \\ & & \| \\ \mathrm{Comb}_4 & \subseteq & \mathrm{Comb}_2 \end{array}$$

FIGURE 1. Subset Relationships of Different Combination Functions.

Theorem 4.6.

1. Comb_4 is a combination function.
2. Furthermore, for all theories $T_1, \ldots, T_n$ and any set IC of integrity constraints, $\mathrm{Comb}_4(\{T_1, \ldots, T_n\}, IC) \subseteq \mathrm{Comb}_1(\{T_1, \ldots, T_n\}, IC)$. □

An important point to note is that there is no obvious "subsetlike" relationship of the sort proved in Theorem 4.6 that links Comb_3 to Comb_4. The following two examples confirm this. Example 4.8 shows that it may not always be the case that $\mathrm{Comb}_3(\{T_1,T_2\},IC) \subseteq \mathrm{Comb}_4(\{T_1,T_2\},IC)$, and Example 4.9 demonstrates that the reverse inclusion does not hold either.

Example 4.8. Suppose T_1,T_2,IC are as given below:

$T_1 = \{\neg a,\ a \vee b,\ \neg c\}$

$T_2 = \{\neg b,\ a \vee b\}$

$IC = \{\neg c\}$

In this case, $\mathrm{MAXCONS}(T_1 \cup T_2) = \{\{a \vee b, \neg c, \neg b, a \vee c\}, \{\neg a, \neg b, \neg c\}, \{\neg a, a \vee b, \neg c\}\}$ $\mathrm{Comb}_3(\{T_1,T_2\},IC) = \mathrm{MAXCONS}\ (T_1 \cup T_2)$.

However, $\mathrm{Comb}_4(\{T_1,T_2\},IC) = \{\{a \vee b), \neg c, \neg(b, a \vee c\}\}$.

Thus, in this case, $\mathrm{Comb}_3(\{T_1,T_2\},IC) \not\subseteq \mathrm{Comb}_4(\{T_1,T_2\},IC)$. □

Example 4.9. Suppose T_1,T_2,IC are as given below:

$T_1 = \{(a \vee b)\}$

$T_2 = \{\neg a\}$

$IC = \{\neg b\}$

In this case, $\mathrm{Comb}_3(\{T_1,T_2\},IC) = \varnothing$ whereas $\mathrm{Comb}_4(\{T_1,T_2\},IC) = \{\{(a \vee b)\},\{\neg a\}\}$.

Thus, this example demonstrates that $\mathrm{Comb}_4(\{T_1,T_2\},IC)$ may not be a subset of $\mathrm{Comb}_3(\{T_1,T_2\},IC)$. □

We say that combination function Comb_i is a *subset* of Comb_j, denoted $\mathrm{Comb}_i \subseteq \mathrm{Comb}_j$, iff, for all theories $T_1, \ldots, T_n$ and integrity constraints IC, it is the case that

$$\mathrm{Comb}_i(\{T_1, \ldots, T_n\},IC) \subseteq \mathrm{Comb}_j(\{T_1, \ldots, T_n\},IC).$$

Figure 1 shows the complete "subset" relationships between the four different combination functions we have defined so far. All subset relationships are shown in Figure 1; the absence of a subset relationship means it does not hold.

From the point of view of ease of implementation, we believe that Comb_1 and Comb_2 are the hardest to implement, while Comb_3 and Comb_4 are relatively easier to implement. Suppose $T_1, \ldots, T_n$ are the first-order theories and IC is a set of integrity constraints. In the case of Comb_3, we find the maximal consistent subsets of $(T_1 \cup \cdots \cup T_n)$ and then simply delete those that violate the integrity constraints.

In the case of Comb_2 and hence of Comb_1, we need to perform some additional work. Those maximal consistent subsets that are not deleted when computing Comb_3 must be further examined: Suppose that Y_i is one such maximally consistent subset of $(T_1 \cup \cdots \cup T_n)$ that violates IC. Then we need to find maximal subsets of Y_i that are consistent with IC. This may involve a great deal of additional work.

Implementing Comb_4 is different. We compute maximally consistent subsets of $(T_1 \cup \cdots \cup T_n \cup IC)$ that contain IC. It is easy to see that this is of the same order of complexity as computing the maximal consistent subsets of $(T_1 \cup \cdots \cup T_n)$. We then simply retain those maximal subsets that are of the greatest cardinality. A detailed discussion of the complexity of these problems, however, is beyond the scope of the paper.

4.2 Relationship with Updating of Theories

The problem of updating theories and revising beliefs has been studied extensively (Fagin, Ullman, and Vardi 1986, Fagin, Kuper, Ullman, and Vardi 1986, Gardenfors 1988, Dalal 1988, Katsuno and Mendelzon 1989, Gardenfors and Makinson 1988, and Satoh 1988). In brief, an update may take one of two forms:

1. An insertion: In this, some *new* information is added to T.
2. A deletion: Here, either a formula F in T is deleted, or a formula entailed by T is deleted.

Inserting a formula F into T may lead to an inconsistency. In this case, the *result* of the insertion must be defined in such a way that the inconsistency is properly handled. Deletions do not give rise to inconsistencies (unless T was already inconsistent).

There is some resemblance between insertions and the combination of theories we have discussed earlier. Insertions into theories may be viewed as a special case of our framework: Take the sentence F to be inserted to be an integrity constraint, that is, $IC = \{F\}$ and now compute MAXCONS(T,IC). This is the gist of Theorem 4.7 below.

One may wonder whether *combining* theory T_1 with theory T_2 may be accomplished by inserting each element of T_2 into T_1. This is not true in general (see Example 4.10 below). The reason for this is that we do not have any priorities over the set of the combined theories.[4] For example, when we insert elements of T_2 into T_1 one by one, then the last element of T_2 to be inserted is accorded the status of an integrity constraint, even though this last element of T_2 may not be present in all maximal consistent subsets of $T_1 \cup T_2$.

Fagin, Ullman, and Vardi (1983) present a theory of updating theories. Before discussing the relationship between updating and combining, we discuss the theory-updating approach of Fagin, Ullman, and Vardi (1983).

Definition 4.10 (Insertion and Deletion) (Fagin, Ullman, and Vardi 1983). Let T be a theory and T^* be the set of clauses logically implied by T. A theory S is said to accomplish the *insertion* of a clause σ into T if $\sigma \in S$. A theory S is said to accomplish the *deletion* of a clause σ into T if $\sigma \notin S^*$.

Definition 4.11 (Fagin, Ullman, and Vardi 1983). Let T, T_1, and T_2 be theories. T_1 has *fewer insertions* than T_2, if $T_1 - T \subset T_2 - T$. T_1 has *no more insertions* than T_2, with respect to T, if $T_1 - T \subseteq T_2 - T$. T_1 has the *same insertions* as T_2, with respect to T, if

[4] Using Gärdenfors's terminology, our combination functions do not satisfy axiom K_2^+ (Gardenfors 1988, p. 48).

$T_1 - T = T_2 - T$. T_1 has *fewer deletions* than T_2, if $T - T_1 \subset T - T_2$. T_1 has *no more deletions* than T_2, with respect to T, if $T - T_1 \subseteq T - T_2$. T_1 has the *same deletions* as T_2, with respect to T, if $T - T_1 \subseteq T - T_2$.

Definition 4.12 (Fagin, Ullman, and Vardi 1983). Let T, T_1, and T_2 be theories. T_1 accomplishes an update u (could be an insertion or a deletion) of T *with a smaller change* than T_2 if both T_1 and T_2 accomplish u, and either T_1 has fewer deletions than T_2 or T_1 has the same deletions as T_2 but T_1 has fewer insertions than T_2.

Definition 4.13 Minimal Updating (Fagin, Ullman, and Vardi 1983). A theory S accomplishes an update u of a theory T *minimally* if there is no theory S' that accomplishes u with a smaller change than S.

In the following theorem, Fagin, Ullman, and Vardi (1983) present a different characterization of minimal updating of theories. This new characterization can be used to construct algorithms that can minimally update a theory.

Theorem 4.7 (Fagin, Ullman, and Vardi 1983). Let S and T be theories and let σ be a sentence. Then,

1. S accomplishes the deletion of σ from T minimally if and only if S is a maximal subset of T that is consistent with $\neg\sigma$, and
2. S accomplishes the insertion of σ from T minimally if and only if $S \cap T$ is a maximal subset of T that is consistent with σ. □

The following theorem describes a relationship between combining theories and updating theories when the union of the theories to be combined is consistent.

Theorem 4.8. Let $T_1, \cdots, T_k$ be theories to be combined in the presence of a finite set IC of integrity constraints. Let σ_{ic} be the conjunction of the integrity constraints in IC. If $T_1 \cup \cdots \cup T_k$ is consistent then, $\text{Comb}_1(\{T_1, \cdots, T_k\},\{\sigma_{ic}\}) = \{X|X$ accomplishes the insertion σ_{ic} into $T_1 \cup \cdots \cup T_k$ minimally.$\}$ □

Proof. $[\text{Comb}_1(\{T_1, \cdots, T_k\},\sigma_{ic}) \subseteq \{X|X$ accomplishes the insertion of σ_{ic} into $T_1 \cup \cdots \cup T_k$ minimally$\}]$. Suppose $X \in \text{Comb}_1(\{T_1, \cdots, T_k\},\sigma_{ic})$. Then X is a maximally consistent set such that $\{\sigma_{ic}\} \subseteq X \subseteq T_1 \cup \cdots \cup T_k \cup \{\sigma_{ic}\}$. Hence, $\sigma_{ic} \in X$. It now suffices to show that X is a maximally consistent subset of $T_1 \cup \cdots T_k \cup \{\sigma_{ic}\}$. But this is true. Hence $X \in \{X|X$ accomplishes the insertion of σ_{ic} into $T_1 \cup \cdots \cup T_k$ minimally$\}$.

$[\{X|X$ accomplishes the insertion of σ_{ic} into $T_1 \cup \cdots \cup T_k$ minimally$\} \subseteq \text{Comb}_1(\{T_1, \cdots, T_k\},\sigma_{ic})]$. The proof is similar. □

Theorem 4.8 above demonstrates that the Fagin, Kuper, Ullman, and Vardi (1986) framework for inserting sentences into theories may be captured in our framework. The example below shows that successive insertions of sentences of theory T_2 into theory T_1 does not correctly capture the combination of theories.

Example 4.10. Suppose

$$T_1 = \{a \rightarrow c,\ a \rightarrow \neg b,\ a \rightarrow b\},$$

and $T_2 = \{a\}$. Inserting a into T_1 causes inconsistency as $a \rightarrow \neg b$ and $a \rightarrow b$ entail $\neg a$. There are two sets that accomplish the insertion:

$$S_1 = \{a, a \rightarrow c, a \rightarrow \neg b\},$$

and

$$S_2 = \{a, a \rightarrow c, a \rightarrow b\},$$

c is true in both S_1 and S_2. Note, however, that, according to MAXCONS($T_1 \cup T_2$), c is not true in T_1, which is maximally consistent of $T_1 \cup T_2$. □

We now use the theory developed by Fagin, Kuper, Ullman, and Vardi (1986) for updates in flocks to compare with our problem of combining theories.

Definition 4.14 Minimal Flock Updating (Fagin, Kuper, Ullman, and Vardi 1986). Let $S' = \{S_1, \cdots, S_n\}$ be a flock. A flock $\mathcal{T} = \{T_1, \cdots, T_n\}$ accomplishes an update u of S' minimally if T_1 accomplishes the update of S_1 minimally. □

Fagin, Kuper, Ullman, and Vardi (1986) defined the updating of a flock as follows:

Definition 4.15 (Fagin, Kuper, Ullman, and Vardi 1986). Let S' be a flock and $S'_1, \cdots, S'_k$ be the flocks that accomplish an update u of S' minimally. Then the result of u is the flock $\bigcup_{1 \leq i \leq k} S'_i$. □

We would like to change Definition 4.15 so that the resulting flock consists of maximal elements only. Formally,

Definition 4.16. Let S' be a flock and $S'_1, \cdots, S'_k$ be the flocks that accomplish an update u of S' minimally. Then the result of u is the flock consisting maximal elements of $\bigcup_{1 \leq i \leq k} S'_i$. □

The following result relating combining theories with updating flocks of theories is immediate. It uses the modified definition of updating a flock as defined in definition 4.16.

Theorem 4.9. Let $T_1, \cdots, T_k$ be a set of theories and let IC be a finite set of integrity constraints. Let $\mathcal{T}$ be the flock MAXCONS($T_1 \cup \cdots \cup T_k$). Let σ_{ic} be the conjunction of the integrity constraints in IC. $\text{Comb}_1(\{T_1, \cdots, T_k\}, \{\sigma_{ic}\})$ = the flock obtained by updating $\mathcal{T}$ with σ_{ic} by using definition 4.16. □

5. COMBINING PRIORITIZED THEORIES

The different knowledge bases that have to be combined might have different priorities associated with them. Intuitively, there might be compelling reasons that cause one knowledge base to be preferable to another. In such a case, we would like to use this priority information while combining the knowledge bases. For example, if a knowledge base with higher priority or believability directly contradicts another knowledge base with a lower priority with respect to a certain aspect, we might want the combined theory to contain the point of view of the knowledge base with the higher priority. In this section, we formalize what we mean by combining prioritized theories.

In the context of the murder example of section 2, the information provided by Bill (the younger witness who was also closer to the scene of action) may seem more credible to the police who may then pursue further investigations based on his version of events, as opposed to John's version.

As a first step, suppose we have theories $T_1, \cdots T_k$ to be combined with the priority relation (a total order) where the priority of T_i is less than the priority of T_j iff $i < j$. With a slight abuse of notation, it can be written as, $T_1 < T_2 < \cdots < T_k$. As always, the integrity constraints have the highest priority, that is, $T_k < IC$. There are two distinct approaches to combine the theories that come to mind immediately: *bottom-up* and *top-down*.

In the bottom-up approach, we start by combining T_1 and T_2 with preference to T_2. The combined theory is then a set of theories defined as $\{T : T_2 \subseteq T$ and $T - T_2$ is a maximal subset of T_1 such that T is consistent$\}$. The result is then combined with T_3 with preference to T_3. This means that each theory in the result is combined with T_3, and the final result is the set that is the union of particular results (each result is a set). This continues until T_k. The result is then combined with IC with preference to IC.

Algorithm 5.1 Procedure. Comb—botup (Comb$_i$, $T_1 \preccurlyeq \cdots \preccurlyeq T_k$,$IC$)

$T_{k+1} = IC$
Comb $= \{T_1\}$
For $i = 1$ to k do
  begin
  Temp $= \varnothing$
  For all T in Comb do
    Temp $=$ Temp $\cup$ Comb$_i$($\{T\}$,T_{i+1})
  Comb $=$ maximal elements of Temp
  end
Output(Comb)

In the top-down approach, we start with combining T_k and IC with preference to IC. The result is combined with T_{k-1} with preference to the theories in the result. This is continued until T_1.

Algorithm 5.2 Procedure. Comb—topdn(Comb$_i$,$T_1 \preccurlyeq \cdots \preccurlyeq T_k$,$IC$)

$T_{k+1} = IC$
Comb $= \{T_{k+1}\}$
For $i = k$ to 1 do
  begin
  Temp $= \varnothing$
  For all T in Comb do
    Temp $=$ Temp $\cup$ Comb$_i$($\{T_i\}$,T)
  Comb $=$ maximal elements of Temp
  end
Output(Comb)

The following example demonstrates the difference between bottom-up and top-down combining.

Example 5.1 ("Flying Snakes"). Consider the theories T_1,T_2,T_3 and the set IC of integrity constraints given below: The priority ordering we have is $T_1 < T_2 < T_3 < IC$.

$$T_1 = \{\neg\text{have\_wings(snakes)}\}$$
$$T_2 = \{\neg\text{have\_wings(snakes)} \rightarrow \neg\text{fly(snakes); fly(snakes)}\}$$
$$T_3 = \{\neg\text{fly(snakes)}\}$$
$$IC = \emptyset$$

Thus, if we consider T_1,T_2,T_3 to have been provided by three different experts E_1,E_2,E_3, the ordering $T_1 < T_2 < T_3$ denotes that we have more faith in E_2 than in E_1 and we have the greatest degree of belief in E_3. E_1 tells us that snakes don't have wings. E_2 tells us two things: first, that "things that don't have wings don't fly" and he also tells us that "snakes fly."[5] T_3 tells us that "snakes don't fly." Bottom-up and top-down combination (using Comb_1) with priorities leads to two different results:

Top-down:
Combining T_3 and T_2, we obtain

$$T_{32} = \{\{\neg\text{have\_wings(snakes)} \rightarrow \neg\text{fly(snakes)}; \neg\text{fly(snakes)}\}\}.$$

Combining T_{32} with T_1, we obtain

$$T_{321} = \{\{\neg\text{have\_wing(snakes)} \rightarrow \neg\text{fly(snakes)};\neg\text{fly(snakes)};\neg\text{have\_wings(snakes)}\}\}.$$

Bottom-up:
Combining T_1 and T_2, we obtain

$$T_{12} = \{\{\neg\text{have\_wings(snakes)} \rightarrow \neg\text{fly(snakes); fly(snakes)}\}\}.$$

Combining T_{12} with T_3, we obtain

$$T_{123} = \{\{\neg\text{have\_wings(snakes)} \rightarrow \neg\text{fly(snakes)}; \neg\text{fly(snakes)}\}\}.$$

Hence, in this case, the result obtained by the top-down approach is different from the result obtained by the bottom-up approach.

An (artificial) objection may be raised that our assignment of priorities may actually be inappropriate because expert E_2 provides the patently false piece of information that snakes can fly. However, this objection assumes that we assign priorities to the experts' credibility *after* they provide their advice. Furthermore, in this example, it is easy to see that the statement "snakes can fly" is false because all of us know something about snakes. On the other hand, if the experts were providing information about a domain of which the reader had little knowledge (such as the history of ancient Egypt, say), he may make a patently false statement such as "Tutankhamen was Pharaoh before Akhenaton" without this being readily identifiable as a falsehood. Finally, it may be possible that the proposition "snakes fly" is not given as an objective fact by expert E_2, but is derived from a collection of a thousand rules (not listed in T_2 above), of which 999 are correct, and one is incorrect. In such cases, it may make perfect sense to prefer E_2's advice over E_1's because only one of 1000 rules provided by him is flawed. □

[5]Our example would be more forceful if we actually modify T_2 so that expert E_2 tells us that "all things that don't have wings don't fly" and then goes on to give a long list of things that don't have wings—for example, cats, dogs, lizards, crocodiles, elephants, etc. However, physically describing the combination of these theories would then be more cumbersome, and this is why we restrict ourselves to this small example.

Example 5.2 (Murder Example Revisited). Let us revisit the murder example. Here, we may give priority to the facts reported by Bill instead of those reported by Ed because Bill was closer to the scene of the murder, and, in addition, he has better eyesight (as he is much younger). In this case, both bottom-up and top-down combinations result in the same conclusions: one where Bill's first statement ("the murderer wore a dark coat") is discarded, and everything else is accepted. □

In general, top-down combining is more informative than bottom-up combining. This is because top-down combining results in a combination with more information than the bottom-up approach. This is evident from the "flying snakes" example. In general, if we are combining theories $T_1, \ldots, T_n$ and integrity constraints IC, what may happen in the bottom-up combination is the following. We start by combining T_1 and T_2 to get a new theory T_{12}. In the process, we may discard a wff W_1 from T_1 because it is inconsistent with a wff or set of wffs in T_2. When we combine the T_{12} with T_3 (call the result T_{123}), we may discard that part of T_2 that causes W_1 to be discarded. The combination of T_{12} would not, then, restore W_1 even though W_1 may be consistent with the new combination T_{123}. Thus W_1 would be "lost."

But why then consider both approaches? It is because both approaches have parallels in real-life decision making. The bottom-up approach corresponds to the following scenario. Consider an organization with a strict hierarchical employee structure. When the employee at the bottom of the hierarchy proposes an idea to his supervisor, his supervisor combines the employee's idea with his beliefs (with more priority to his beliefs) and passes it on to his supervisor. This goes on till it reaches the highest authority. This passing of an idea corresponds to the bottom-up approach. In fact, this mimics precisely what happens in bottom-up combination: Employee E_1's supervisor E_2 may never pass on some of E_1's input to his supervisor E_3. Thus, that input would never trickle up the hierarchy and would be "lost." Similarly, the top-down corresponds to the case when the topmost manager passes an idea or order to his subordinate and he to his subordinate and so on.

Coming back to the problem of updating, one may wonder whether the *combining* of prioritized theories is similar to the updating problem. Since the motivations for the two problems are different, so are the results. The updating problem can be simulated by the *bottom-up* procedure for combining *prioritized* theories, which is different from both the *top-down* procedure and the procedure for combining nonprioritized theories.

While speaking of the combination of prioritized theories, we briefly mention that it is possible that we have prioritized groups of theories $G_1, \ldots, G_n$. Any two theories in the same group have the same priority; however, the groups themselves have priorities $G_1 < G_2 < \cdots < G_n$. The most straightforward way of combining the resulting multitude of theories is to proceed as follows:

Algorithm 5.3. Combining Prioritized Groups of Theories

Step 1. For all $1 \leq i \leq n$, set $S_i = \text{MAXCON}({}_{T \in G_i} T)$. Thus, at this stage, for any $1 \leq i \leq n$, all theories in the group G_i have been combined together. Each S_i is thus a set of theories.

Step 2. Construct $S = (S_1 \times \cdots \times S_n)$, that is, S is the Cartesian product of the S_is.

Step 3. Let $S = \{V_1, \ldots, V_k\}$ where each V_i is of the form $\langle s_1, \ldots, s_n \rangle$ where $s_j \in S_j$ for all $1 \leq j \leq n$.

Step 4. Combine theories $s_1, \ldots,$ with priorities $s_1 < s_2 < \cdots < s_n$. Do this for all $V_i \in S$. Let the resulting theories be $\{TH_1, \ldots, TH_k\}$.

Step 5. Choose the maximal elements of $\{TH_1, \ldots, TH_k\}$.

A detailed discussion about combining prioritized groups of theories is beyond the scope of this paper.

In order to see how the work described thus far relates to existing work on reasoning with inconsistency, consider the following program P:

p
$\neg p$
$p \rightarrow q$
$\neg p \rightarrow q$
p and $\neg p \rightarrow r$

Using the annotated logic semantics of Blair and Subrahmanian (1988, 1989) and later improved by Kifer and Lozinskii (1989), it would be possible to infer r even though r depends on a somewhat shaky justification: (p and $\neg p$). However, the annotated logic semantics does not allow us to conclude $\neg r$ or $\neg q$. Clearly, the fact that $\neg r$ and $\neg q$ cannot be inferred corresponds quite well with our intuition. The MAXCONS(P) approach would allow us to conclude r. At the same time, neither $\neg r$ nor $\neg q$ can be concluded.

Considering the same program P, the semantics of Gelfond and Lifschitz (1990) and Kowalski and Sadri (1990) allows us to conclude everything. In particular, r can be concluded (in the same way as in Blair and Subrahmanian (1988, 1989), but in addition, both $\neg r$ and $\neg q$ may be concluded.

One advantage of the Blair and Subrahmanian (1988, 1989; Kifer and Lozinskii 1989) approach is that, in the case of programs containing function symbols, their semantics leads to a semidecidable consequence relation. When fuction symbols are present, this does not appear to be true for the MAXCONS(P) semantics.

6. DISCUSSION AND CONCLUSIONS

Expert database systems are traditionally built by a team of knowledge engineers who elicit knowledge from multiple experts in the domain of interest. Each of these experts provides information (which we will assume is represented in first-order logic). As the people being asked for information are experts in the domain of interest, one would usually assume that the information they provide is correct and usually this is the case. However, two phenomena that makes the problem of coalescing the information provided by multiple experts much harder are:

An expert provides some information that is incorrect (false). In such a case, it may be possible to detect and correct the error; the expert will usually be quick to change his/her mind when the mistake is pointed out. However, the database may be in use for quite some time before the mistake is ever detected.

Two experts may *disagree*. In this case, there is no easy way to rectify the problem. The disagreement of the two experts manifests itself as an inconsistency, and one is forced to continue reasoning despite the presence of the inconsistency.

Hence, the process of combining multiple knowledge bases is a genuine problem that arises frequently. In this paper, we have provided four ways of combining multiple knowledge bases when these knowledge bases all have the same priority, that is, they are all considered to have been obtained from equally credible sources. These four ways are based on the functions Comb_1,Comb_2,Comb_3, and Comb_4. We prove various relationships between these four combination techniques. All these techniques deal with methods to manipulate "maximal" consistent subsets. The differences arise depending upon:

Step 1. Which is the set of formulas whose maximal consistent subsets are being considered? and

Step 2. Having made a selection in the previous step (above), which maximal consistent subsets do we retain as "acceptable" and which do we discard?

It turns out that $Comb_1$ and $Comb_2$ give the same results, but compute the results differently. $Comb_3$ and $Comb_4$ are "bolder" than $Comb_1$ in the sense that they may discard more maximal consistent subsets in step 2 than $Comb_1$ does. $Comb_3$ acts on the philosophy that the integrity constraints (if any) play a role only in determining which maximal consistent subsets should be chosen. Thus, in step 1, $Comb_3$ ignores the integrity constraints and finds various maximal subsets and then, in step 2, discards some (possibly none) of the maximal consistent subsets. The great danger with $Comb_3$ is that it may choose to discard everything generated in step 1. $Comb_4$, on the other hand, never discards everything. What it does is to simply pick all of the maximal consistent subsets of the highest possible cardinality. As stated earlier, the intuitive motivation behind $Comb_4$ is that "every time we discard something, we are (implicitly) saying that the discarded object constitutes a mistake made by the expert. As this information is obtained from experts, it may be advisable to assume that they don't make too many mistakes; hence, we want to pick those maximal consistent subsets where the number of mistakes is minimized." $Comb_4$ thus seems to be suitable in situations where it is appropriate to make this assumption. $Comb_3$, on the other hand, is more appropriate in situations where the integrity constraints do not play a role in computing maximal consistent subsets. This may arise if the experts are providing advice about a domain, but they are not fully aware of some of the constraints that the individual seeking the advice has. For instance, in an expert system for putting out oil fires that may be used by many different companies engaged in this task, a user may have the constraint "Our company can make an initial financial outlay of only $274,000 towards fighting this fire." This is a constraint that the experts don't have. However, this constraint limits the company in the range of choices it can adopt. If the maximal consistent subsets correspond to the range of choices available, then this user constraint limits the choices the company has to those maximal consistent subsets that satisfy the $274,000 constraint. Thus, $Comb_3$ may find applications in situations where the user (not the expert) keeps adding additional constraints. $Comb_1$, which coincides with $Comb_2$ is the most conservative of all the combination functions. It generates all possibilities and does not ignore any piece of information. It is an appropriate combination function to use in situations where it is essential to consider all possibilities (such as in life-threatening situations and in medical expert systems).

$Comb_1$,$Comb_2$,$Comb_3$,$Comb_4$ all assume that the experts providing different data are all equally credible. This is not always an appropriate assumption to make because we often have different "degrees of faith" in the credibility of the experts providing information. For instance, we may give greater weight to an economic forecast made by a Nobel Laureate in economics than by a graduate student in economics. When we have a linearly ordered set of theories (where the linear order reflects the prioritization of the theories), we have presented two prioritized combination schemes top-down and bottom-up. In the bottom-up approach, some information may be "lost," while in the top-down approach, this information may not be lost. Nonetheless, the bottom-up approach, despite the loss of information, mirrors the way strict administrative hierarchies, such as in companies, often work; some information/ideas from the lower rungs of the hierarchy are "lost" when they propagate upward.

These are some of the guidelines that a database designer may choose when deciding which algorithms to use. It is really up to the individual researcher to decide which semantics are most appropriate for his/her use. We do not believe that there is one all-encompassing way of combining multiple knowledge bases.

We are currently extending this work in different directions. One direction is related to the question "How should we combine multiple knowledge bases consisting, not of first-order theories, but nonmonotonic theories like default theories or autoepistemic theories?". Another important direction is a kind of hybrid reasoning: How do we integrate knowledge bases expressed in different formalisms?

ACKNOWLEDGMENTS

The authors wish to express their appreciation to the National Science Foundation for support of their work under grant numbers IRI-86-09170 and IRI-91-09755 as well as the Army Research Office under grant number DAAG-29-85-K-0-177. Sarit Kraus was partially supported by National Science Foundation grant IRI-8907122. Thanks go to Daniel Lehmann and to the referees for providing useful comments. The paper has been strengthened by their constructive criticism. Thanks to Lev Novik for helping with the diagram.

REFERENCES

BARAL, C., S. KRAUS, and J. MINKER. 1989. Combining multiple knowledge bases. Technical Report UMIACS TR 89–90, CS TR 2316, Department of Computer Science and UMIACS, University of Maryland, College Park, MD, September 1989. Also in IEEE transactions on data and knowledge engineering, 3(2): 200–220, June 1991.

———. 1990. Communicating between multiple knowledge-based systems with different languages. *In* S. M. Deen, editor, Proceedings of the Working Conference on Cooperating Knowledge Based Systems, England, p. 121–124.

BLAIR, H. A. and V. S. SUBRAHMANIAN. 1988. Paraconsistent foundations for logic programming. Journal of non-classical logic, 5(2): 45–73.

———. 1989. Paraconsistent logic programming. Theoretical computer science, 68:135–154.

DA COSTA, N. C. A., V. S. SUBRAHMANIAN, and C. VAGO. 1991. The paraconsistent logics $\mathcal{P}T$. Zeitschrift fur Mathematische Logik und Grundlagen der Mathematik, 37: p. 139–148.

DALAL, M. 1988. Investigations into a theory of knowledge base revision, preliminary report. *In* AAAI, p. 475–479.

ETHERINGTON, D. 1988. Reasoning with incomplete information. Morgan Kaufmann Publishers, CA.

FAGIN, R., G. KUPER, J. ULLMAN, and M. VARDI. 1986. Updating logical databases. *In* Advances in Computing Research, Vol. 3, p. 1–18.

FAGIN, R., J. D. ULLMAN, and M. Y. VARDI. 1983. On the semantics of updates in databases. *In* ACM SIGACT/SIGMOD symposium on principles of database systems, p. 352–365.

GARDENFORS, P. 1988. Knowledge in flux: modeling the dynamics of epistemic states. MIT Press.

GARDENFORS, P. and D. MAKINSON. 1988. Revisions of knowledge systems using epistemic entrenchment. *In* Proceedings 2nd Conference on Theoretical Aspects of Reasoning about Knowledge. Pp. 83–95.

GELFOND, M. and V. LIFSCHITZ. 1990. Logic programs with classical negation. *In* Proceedings 7th International Conference on Logic Programming. MIT Press. Pp. 579–597.

GRANT, J. 1974. Incomplete models. Notre Dame journal of formal logic, XV(4):601–607.

———. 1975. Inconsistent and incomplete logics. Mathematics magazine of mathematical association of america, 48(3):154–159.

———. 1977. Incompleteness and inconsistency in propositional logic. Relevant logic newsletter.

———. 1978. Classifications for inconsistent theories. Notre Dame journal of formal logic, XIX(3):435–444.

GRANT, J. and V. S. SUBRAHMANIAN. 1990. Reasoning in inconsistent databases. Manuscript.

KATSUNO, H. and A. MENDELZON. 1989. A unified view of propositional knowledge base updates. *In* Eleventh International Joint Conference on AI, Detroit, August 1989, p. 1413–1419.

KIFER, M. and E. LOZINSKII. 1989. A logic for reasoning with inconsistency. To appear in Journal of automated reasoning. Preliminary version in Proceedings Logic in computer science, pp. 253–262.

KOWALSKI, R. and F. SADRI. Logic programs with exceptions. *In* Proceedings 7th International Conference on Logic Programming. MIT Press. Pp. 598–613.

PLAISTED, D. 1984. Complete problems in the first order predicate calculus. Journal of computer and systems sciences, 29:8–35.

REITER, R. 1980. A logic for default reasoning. Artificial intelligence, 13:81–132.

KEN SATOH. 1988. Non-monotonic reasoning by belief revision. *In* Proceedings of the International conference on fifth generation computer systems. Pp. 455–462.

SUBRAHMANIAN, V. S. 1992. Paraconsistent disjunctive deductive databases. To appear in Theoretical computer Science.

VARDI, M. 1982. The complexity of relational query languages. Proceedings ACM symposium on theory of computing, pp. 137–146.

APPENDIX: DESCRIPTION OF ALGORITHMS

Algorithm A.1 Procedure. MAXCONS1(P)

MAXCON = $\varnothing$
If P in consistent, then MAXCONS1(P) = $\{P\}$.
else
begin
{** Let P be the set of sentences $\{C_1, \cdots, C_n\}$. **}
For $i = 1 \cdots n$ do $P_i := P - \{C_i\}$ od
For $i = 1 \cdots n$ do MAXCON:= MAXCON $\cup$ MAXCONS1(P_i) od
MAXCONS1(P) := maximal elements of MAXCON.
end

Algorithm A.2 Procedure. MAXCONS1(P,IC)

MAXCON = $\varnothing$
If $P \cup IC$ is consistent, then MAXCONS1(P,IC) = $\{P\}$.
else
begin
{** Let P be the set of sentences $\{C_1, \cdots, C_n\}$. **}
For $i = 1 \cdots n$ do $P_i := P - \{C_i\}$ od
For $i = 1 \cdots n$ do MAXCON:= MAXCON $\cup$ MAXCONS1(P_i,IC) od
MAXCONS1(P,IC):= maximal elements of MAXCON.
end

Algorithm A.3 Procedure. MAXCONS2(P)

Let there be n potential causes of the inconsistency of P.
If $n = 0$, then MAXCONS2(P) = $\{P\}$.
else
begin

```
For i = 1 ··· n do S_i:= the ith potential cause of the inconsistency of P od
S:= S_1 × ··· × S_n
S':= {{a_1, . . . , a_n} | (a_1, . . . , a_n) ∈ S}
MINS:= minimal elements of S' WRT inclusion
MAXCONS2(P) = {P − Y. | Y ∈ MINS}
end
```

Algorithm A.4 Procedure. MAXCONS2(P,IC)

```
Let there be n potential causes of the inconsistency of P ∪ IC
If n = 0, then MAXCONS2(P,IC) = {P}.
else
begin
For i = 1 ··· n do S_i:= (the ith potential cause of the inconsistency) - IC od
S:= S_1 × ··· × S_n
S':= {{a_1, . . . , a_n} | (a_1, . . . , a_n) ∈ S}
MINS:= minimal elements of S' WRT inclusion
MAXCONS2(P,IC) = {(P − Y) | Y ∈ MINS}
end
```

Algorithm A.5 Procedure. MAXCONS3(P,IC)

```
MAXCON = ∅.
CHECK = {P}.
Found = false
While not Found
 begin
TEMP = ∅.
  For all P_i ∈ CHECK do
    begin
    If P_i ∪ IC is consistent then
      begin
      Found=True.
      MAXCON = MAXCON ∪ {P_i ∪ IC}.
      end
    else
      begin
      {** Let P_i be the set of sentences {C_i1, ··· , C_in. **}
      For j = 1 ··· n do TEMP = TEMP ∪ {P_i − {C_ij}} od
      end
    end CHECK = TEMP.
  end
MAXCONS3(P,IC):= maximal elements of MAXCON.
end
```