Assumption Based Reasoning and Clause Management Systems

by

Alex Kean and George Tsiknis

Technical Report 90-9
May, 1990

Department of Computer Science
University of British Columbia
Vancouver, British Columbia
V6T 1W5
Canada

Assumption Based Reasoning and Clause Management Systems

ALEX KEAN and GEORGE TSIKNIS

*Department of Computer Science, University of British Columbia,
Vancouver, British Columbia V6T 1W5, Canada*

Technical Report 90-9

March 19th 1990

A *truth maintenance system* is a subsystem that manages the utilization of assumptions in the reasoning process of a problem solver. Doyle's original motivation for creating a truth maintenance system was to augment a reasoning system with a control strategy for activities concerning its non-monotonic state of beliefs. Hitherto, much effort has been invested in designing and implementing the concept of truth maintenance and little effort has been dedicated to the formalization that is essential to understanding it. This paper provides a complete formalization of the principle of truth maintenance. Motivated by Reiter and de Kleer's preliminary report on the same subject, this paper extends their study and gives a formal account of the concept of truth maintenance under the general title of *assumption based reasoning*. The concept of assumption based theory is defined and the notions of explanation and direct consequence are presented as forms of plausible conclusion with respect to this theory. Additionally, the concept of extension and irrefutable sentences are discussed together with other variations of explanation and direct consequence. A set of algorithms for computing these conclusions for a given theory are presented using the notion of prime implicates. Finally, an extended example on Boolean circuit diagnosis is shown to examplify these ideas.

1 Introduction

The very idea of assumption based reasoning can be traced back to the work of Bolzano's logic of variations [2]. In the advent of computational machinery, the aspiration of mechanizing reasoning has brought about the revival of such reasoning. To best describe assumption based reasoning, consider the following quote:

> Every proposition is either *true* or *false* and that permanently so. In some cases, however, the same proposition would seem to be at times *true* and at times *false*. The reason for this, according to Bolzano, is that in the original proposition some component, which may not be stated explicitly in the corresponding linguistic expression, has been changed. —— **[Bolzano's Logic'1873 (trans. by Jan Berg 1962), pp 92]**

This is to say that those *changeable* components of the proposition are assumptions which may be true or false at times and they affect the truth of the whole proposition. In human reasoning, it is the ability in keeping score of which assumptions are affecting the conclusion and altering the conclusion as the assumptions are changed that we admire so much.

In a computational framework, the management of these dependency between assumptions and conclusion is supported by a truth maintenance system. A truth maintenance system is usually a subsystem of a problem solver. The original motivation for Doyle [8] in creating a truth maintenance system was to augment a problem solver with control strategy for the activities concerning its non-monotonic state of beliefs represented by assumptions. In the problem solver's knowledge, addition of new information might nullify the validity of some conclusion but justify new ones. The nature of the problem demands the ability of the truth maintenance system to keep track of the relationship between the conclusion and the arguments that justify the conclusion. If some of these arguments are no longer sound due to the addition of knowledge, the conclusion should be denied. This is opposed to the orthodox usage of mathematical logic, where false arguments could be used to conclude universality. In the concept of assumption based reasoning, unsound arguments can be separated and the remaining sound arguments can be used to continue the business of reasoning.

Hitherto, much effort has been invested in designing and implementing the concept of truth maintenance and little effort has been dedicated to the formalization that is essential to its understanding. To this end, Reiter and de Kleer made a first attempt in their 1987 preliminary report on the foundation of truth maintenance systems [15]. This paper, motivated by Reiter and de Kleer's preliminary report, extends this study and gives a formal account of the principle of truth maintenance under the general title of *assumption based reasoning.*

Foremost, there are some important criteria that the design of truth maintenance systems must satisfy. Firstly, the design should be formal so that correctness can be shown. Secondly, it should function independent from domain specific knowledge, that is the more domain independent the more widely adaptable it will be for a spectrum of problem solvers.

What constitutes a formal analysis ? In the case of truth maintenance, it is to investigate a correct, natural and formal definition of the methods required to perform truth maintenance tasks in the most general sense. ATMS [6] on the other hand was built based on some informal notion and many clever intuitions to improve efficiency. The end result was a highly customized and complex program, but it is difficult to analyze whether the tradeoff of expressive power for efficiency is justifiable. More specifically, ATMS started with a restricted type of HORN clauses and later, realizing the need for more general expressive power, it was extended with more complicated heuristics [7]. Admittedly by de Kleer [7, pp 196], the choice of which heuristic to use to increase efficiency for a particular problem solver is an art, violating our proposed criteria of independence from domain specific knowledge.

This is not to argue that specialized truth maintenance systems should not be built. On the contrary, specialized truth maintenance systems can be derived from the general system if required. This has the added advantage that the performance and correctness of the derived system can be measured with respect to the general system.

What is the functionality of truth maintenance systems ? As mentioned earlier, a fundamental funtion of truth maintenance systems is to manage the *dependency* between hypotheses and conclusions. To give a formal account of *dependency*, consider a logic system with some facts $\mathcal{F}$ and a sentence G. Suppose there exists some sentences E (subjected to satisfying some constraints) such that $\mathcal{F} \cup E \models G$, then E is said to justify G with respect to $\mathcal{F}$ and the relation $\mathcal{F} \cup E \models G$ is the dependency. More often the constraint on E is to be consistent with $\mathcal{F}$.

In fact, most systems that have used the notion of dependency between the conclusion and its justification have used the above definition directly, or indirectly through procedural interpretation. The RESIDUE system [9], used the above definition directly and called E the *residue*. In Poole's system of theory formation (THEORIST), he used the same definition and labelled the tuple $(\mathcal{F}, E)$ a *scenario* for G [12]. Another such example can be found in Cox and Pietrzykowski [4] definition of *causes*. In a less obvious manner, Martin and Shapiro [11] presented a formal system of belief revision using the notion of *relevance logic*. They also incorporated the above definition in their notion of *origin-set* for a supported well formed formula. In the two most influential implementations of truth maintenance systems, Doyle [8] used the above definition for justification implicitly in his data structure; while in de Kleer's ATMS [6], the notion of a label in a node can be defined using the above definition with extension to assumption as described in [15].

With the definition of dependency, another crucial issue in truth maintenance is how to encode the knowledge base such that the dependency and justification can be accessed easily, as well as how to update the existing dependencies efficiently when a new piece of knowledge is added. Reiter and de Kleer [15] proposed the strategy of compiling the knowledge base into a set of equivalent and minimal sentences called *prime implicates*, finding the justification for a particular goal is then easily computed via set operations. A more extensive study of this technique can be found in [17]. This transforms the problem of updating the knowledge base into the problem of computing the set of prime implicates incrementally. An incremental algorithm for this task is reported in [10].

The next important issue in truth maintenance is the notion of *relevancy.* For epistemic reasons, if $A \rightarrow C$ and $A \wedge B \rightarrow C$, the preferred dependency to keep is the *minimal* dependency $A \rightarrow C$ because it subsumes the other. Another relevent feature of dependency is the notion of assumption based dependency. This is the original motivation of truth maintenance systems in which the task is to aid a problem solver in reasoning with despite changing assumptions. For instance, an explicit set of symbols are designated assumptions and for those dependency in question, only those that are members of the set of assumptions are considered. Actually, there is more to it than merely matching assumption with dependency, there is this whole notion of *assumption based reasoning*[1] that the truth maintenance system is performing. Hereafter, we shall present a framework for *assumption based reasoning* and a computational system call the *Assumption-based Clause Management System* (*ACMS*).

Section 2 presents some preliminary definitions and results for *implicates* and *supports* that are required in the later sections. In section 3 an assumption based theory is introduced, and in section 4 the computation for the assumption based theory using *prime implicates* and *minimal supports* are illustrated by our *ACMS*. Finally, an extended example on Boolean circuit diagnosis is shown in section 5 and the conclusions can be found in section 6.

[1] In [13], the name is used to categorize a wide range of non-monotonic reasoning including default reasoning, commonsense reasoning and etc. However, the name is used in this paper in the context of reasoning with assumptions and to categorize the activity of truth maintenance.

2 Preliminary

We will assume the mathematics of standard propositional logic and its usage [16]. Assuming a language $\mathcal{L}$ with an enumerable set of propositional variables, a finite set of logical connectives and sentences formed using only these variables and connectives. A *literal* is either a positive (x) or a negative variable ($\neg x$). The literals x and $\overline{x}$ are called a pair of *complementary* literals.

Let $S = \{S_1, \ldots, S_n\}$ be a collection of sentences $S_i, 1 \leq i \leq n$. The sentence $\vee S$ denotes the sentence $S_1 \vee \ldots \vee S_n$ and the sentence $\wedge S$ denotes the sentence $S_1 \wedge \ldots \wedge S_n$. For an empty set S, the sentence $\vee S$ denotes *false* (or in set notation, the symbol □) and the sentence $\wedge S$ denotes *true* (or the symbol ■ in set notation).

Given a *clause set* (or simply a clause) $C = \{c_1, \ldots, c_n\}$ such that for all sentences $c_i, 1 \leq i \leq n$ is a literal, a *disjunctive clause* is a sentence $\vee C$ and a *conjunctive clause* is a sentence $\wedge C$. A clause C is *fundamental* if C does not contain a complementary pair of literals and is *non-fundamental* [2] otherwise.

Given a set $N = \{N_1, \ldots, N_n\}$ such that every $N_i, 1 \leq i \leq n$ is a clause, a sentence N in *conjunctive normal form* (CNF) denotes the sentence $(\vee N_1) \wedge \ldots \wedge (\vee N_n)$ and a sentence N in *disjunctive normal form* (DNF) denotes the sentence $(\wedge N_1) \vee \ldots \vee (\wedge N_n)$. Let $A = \{A_1, \ldots, A_n\}$ be a set of sentences, the negation $\overline{A}$ is defined as the set $\{\overline{A_1}, \ldots, \overline{A_n}\}$. The negation of a sentence, for instance $\neg(\vee A)$, is the sentence $\wedge(\neg A)$[3].

A set of sentences A *subsumes* another set B if $A \subseteq B$. The function $SUB(\Sigma)$ is a subset of Σ such that every sentence in $SUB(\Sigma)$ is not subsumed by another sentence in Σ.

2.1 Implicates

The notion of implicants (the dual of implicates) has been studied extensively in the switching theory literature [1]. In [15], Reiter and de Kleer have exploited the power of prime implicates in an attempt to formalize ATMS. In [17], we have further explored the intricacy of implicates and the following is a summary of their properties. Given a set of sentences Σ in CNF, a disjunctive

[2] A non-fundamental disjunctive clause $\vee C$ is a *tautology* and a non-fundamental conjunctive $\wedge C$ is a *contradiction*.

[3] We shall use the overstrike bar for the negation of a set and "$\neg$" for the negation of a sentence.

clause P is an *implicate* of Σ if $\Sigma \models P$. Thus, a *minimal implicate* is an implicate such that no proper subset of it is an implicate of Σ. We can further categorize a minimal implicate into (i) a *prime implicate* P of Σ is an implicate of Σ such that no other implicate P' of Σ satisfies $\models P' \rightarrow P$; (ii) a non-fundamental disjunctive clause of Σ is a *trivial implicate* of Σ and (iii) a *minimal trivial implicate* of Σ which is a minimal and trivial implicate of Σ.

For convenience, if Σ is a set of sentences in CNF, $MI(\Sigma)$, $PI(\Sigma)$ and $MTI(\Sigma)$ denote the set of all minimal, prime and minimal trivial implicates of Σ in CNF respectively. A disjunctive clause is *entailed* by Σ if there is a minimal implicate in $MI(\Sigma)$ that subsumes it. A similar relationship exists between a fundamental disjunctive clause and $PI(\Sigma)$. Additionally, the sets Σ, $MI(\Sigma)$ and $PI(\Sigma)$ are all logically equivalent in the sense that if a disjunctive clause C is *entailed* by one of the above sets, then the others logically entail C. The inclusion properties among sets of implicates are (i) $PI(\Sigma) \cap MTI(\Sigma) = \emptyset$, (ii) $PI(\Sigma) \subseteq MI(\Sigma)$ and (iii) $MI(\Sigma) = PI(\Sigma) \cup MTI(\Sigma)$.

Methods to compute prime implicates (or dual of prime implicants) are readily available in the literature. Nevertheless, due to the dynamic nature of the reasoning system, there is a need for an incremental prime implicate generator [15]. More precisely, if $\Pi = PI(\Sigma)$ is a set of prime implicates of Σ and C is a disjunctive clause, the problem is defined as computing the revised set of prime implicates of $\Pi \cup \{C\}$. Since the PI operator is non-monotone, it is not advantageous to compute the set $PI(\Pi \cup \{C\})$ using the conventional methods. A new incremental algorithm for this task is studied in [10].

2.2 Support

The notion of a support for a goal or an observation with respect to a knowledge base has played an important role in computational reasoning [4, 9, 12]. In [15], Reiter and de Kleer demonstrated motivations for using the notion of supports, these include aiding abductive reasoning and facilitating search algorithms. Further studies in the intricacy of support are reported in [17] and the following is a summary of the observed properties.

If Σ is a set of sentences in CNF and G is a nonempty disjunctive clause, a disjunctive clause S is a *support* for G with respect to Σ if (i) $\Sigma \models S \vee G$ and (ii) $\Sigma \not\models S$. S is a *minimal support* for G with respect to Σ if no proper subset of S has the same properties. Minimal support is further divided as follows: (i) a *prime support* is a minimal support S such that $S \cup G$ is fundamental and (ii) a *minimal trivial support* is the opposite, i.e. $S \cup G$ is non-fundamental. Finally, the set of supports satisfies the inclusion properties $PS(G, \Sigma) \cap MTS(G, \Sigma) = \emptyset$ and most importantly, $MS(G, \Sigma) = PS(G, \Sigma) \cup MTS(G, \Sigma)$. Methods for computing supports are studied in [15, 17] and the computation of its variants, i.e. direct consequences and explanations, are presented in section 4.

3 Assumption Based Reasoning

We have suggested that assumption based reasoning can provide a formalization of the functionality of truth maintenance systems. Assumption based reasoning is a form of reasoning about conclusions in which the conclusions are affected by assumptions.

Historically, Tarski's classical notion of consequence has prevailed over most modern logical reasoning. The notion of assumption based reasoning was made popular by the AI community in the eighties and has been in existence since the work of Bolzano in 1873 [2, pp 92]. In Bolzano's logic of variation, a conclusion for an argument is always subject to some presuppositions (assumptions). When these presuppositions change, so does the argument for the conclusion. The semantic difference in the notion of consequence between Bolzano and Tarski is beyond the scope of this paper (see [18]).

The purpose of this section is to provide a propositional theory of *assumption based reasoning* in Bolzano's sense (at least in the same spirit as his motivation) but not deviate from the Tarskian semantics by creating a new logic. The eventual goal of course, is to provide a computational framework for assumption based reasoning and to express the functionality of truth maintenance systems.

The knowledge base of a task domain is represented by a set of sentences which are known to be true, the *facts* ($\mathcal{F}$); and a set of sentences called the *assumptions* ($\mathcal{A}$) which represent all the possible hypotheses a reasoner assumes. An *assumption based theory* (α-theory) is a tuple $\mathcal{T} = (\mathcal{F}, \mathcal{A})$ where both $\mathcal{F}$ and $\mathcal{A}$ are well formed sentences in $\mathcal{L}$. The distinction between *facts* and *assumptions* is in the way they are used by the conclusion sanctioning process. From here on, definitions are defined with respect to $\mathcal{T} = (\mathcal{F}, \mathcal{A})$ unless stated otherwise.

3.1 Varieties of Reasoning

Traditionally, logical reasoning in problem solving is closely related to a logical deduction process that associates the conclusions to their premisses by means of sound deductive rules. As the task of problem solving becomes more sophisicated, other type of reasoning know as abduction emerged. We shall introduce a framework for the varieties of abductive and deductive reasoning and when coupled with the notion of assumptions, will form the basis of our proposed assumption based reasoning.

Definition 3.1 *Let $\mathcal{F}$ be a set of sentences (Facts), Ant and Conseq be a sentence respectively, and a relation $\mathcal{R}$ over Ant and Conseq. The conclusions Ant and Conseq are defined respectively as:*

<u>Abduction</u>	<u>Deduction</u>
(1a). $\mathcal{R}(Ant)$ *is true;*	*(2a).* $\mathcal{R}(Conseq)$ *is true;*
(1b). $\mathcal{F} \models Ant \rightarrow Conseq$	*(2b).* $\mathcal{F} \models Ant \rightarrow Conseq$
(1c). $\mathcal{F} \not\models \neg Ant$	*(2c).* $\mathcal{F} \not\models Conseq.$

For (1), if the query is the *Conseq* and the conclusion is the *Ant*, we immediately have a notion of abduction. More precisely, the *Ant* is a consistent hypothesis that sanctions the consequence *Conseq*. The relation $\mathcal{R}(Ant)$ is a constraint on the sentence *Ant* in the abduction. On one extreme, the relation $\mathcal{R}$ can be a theory of constraints and the relation is one of $\mathcal{R} \models Ant$. In the following subsection, we shall explore a variant of abduction by defining $\mathcal{R}$ to be the *subset* relation between *Ant* and a set of assumptions.

Conversely, in (2) if the *Ant* is the query and the *Conseq* is the conclusion, we have a special notion of *deductive consequence.* This definition of deductive consequence says that the answer *Conseq* cannot be concluded from the facts $\mathcal{F}$ alone but it is a logical consequence of facts $\mathcal{F}$ if augmented with *Ant*. Again, the relation $\mathcal{R}$ will play the role of a constraint on *Conseq*. In the latter subsection, we shall present a refinement to this notion of deductive consequence by defining $\mathcal{R}$ to be the *subset* relation between *Conseq* and a set of assumptions.

3.2 Explanations

Finding a consistent hypothesis (or explanation) that sanctions the conclusion is a type of reasoning process that is inevitable in many application domain including diagnostic reasoning. We shall first introduce the notion of explanation; the nature of explainability, agreement and irrefutability; and the concept of assumption based logical extension. The search for an explanation to a question G is a search for a consistent subset of assumptions that together with $\mathcal{F}$, sanction G. We shall call this the *assumption based abductive reasoning.*

Definition 3.2 *Let $\mathcal{T} = (\mathcal{F}, \mathcal{A})$ be an α-theory and the query G be a sentence. A set of sentences E in conjunction (that is, the sentence $\wedge E$) is an explanation*[4] *of G from $\mathcal{T}$ if*

1. $E \subseteq \mathcal{A}$,
2. $\mathcal{F} \cup E \models G$ *and*
3. $\mathcal{F} \cup E$ *is consistent.*

The sentence G is explainable from $\mathcal{T}$ if there exists an explanation of G from $\mathcal{T}$. G is agreeable with respect to $\mathcal{T}$ if G is explainable from $\mathcal{T}$ but its negation is not.

As trivial as the definition is, the following two conditions constitute *inexplicablility*: A query G has no explanation if (i) $\mathcal{F} \cup G$ is inconsistent, that is conditions (2) and (3) are violated and (ii) there exists an E that satisfies (2) and (3) but violates (1). Notice that in the latter case, by varying the assumptions $\mathcal{A}$, a query can be turned from explanable into inexplicable and vice versa.

Generally, a sentence G can have infinitely many explanations if $\mathcal{A}$ is infinite. If E is an explanation of G, any consistent superset of E is also an explanation of G, consequently some minimality restrictions on explanations are required. In addition, it is desirable to distinguish some explanations which trivially entail G independent of any $\mathcal{F}$. We shall therefore introduce the notions of minimality, triviality and primeness of an explanation.

Definition 3.3 *Let $\mathcal{T} = (\mathcal{F}, \mathcal{A})$ be an α-theory , the query G be a sentence and a set of sentences $E \subseteq \mathcal{A}$.*

[4] Throughout the paper, a set of sentences E being an explanation is understood to mean the sentence $\wedge E$.

1. *E is a minimal explanation of G if it is an explanation of G and there is no other explanation E' of G such that E' subsumes E.*
2. *E is a trivial explanation of G if $E \models G$ otherwise E is non-trivial.*
3. *E is a minimal trivial explanation of G if E is both minimal and trivial explanation of G.*
4. *E is a prime explanation of G if it is minimal and non-trivial.*

The following terms $ME(G, \mathcal{T})$, $MTE(G, \mathcal{T})$, $PE(G, \mathcal{T})$ are used to denote the sets of minimal, minimal trivial and prime explanations of a sentence G from an α-theory . It follows trivially from the definitions that the property $ME(G, \mathcal{T}) = PE(G, \mathcal{T}) \cup MTE(G, \mathcal{T})$ holds. Note that assuming consistent $\mathcal{F}$, these minimalities also have the following properties:

1. *If $\mathcal{F} \cup G$ is inconsistent, then $ME(G, \mathcal{T}) = \emptyset$. This is obvious since $\mathcal{F} \models \overline{G}$ therefore $\mathcal{F} \cup E \not\models G$ for any E.*
2. *If G is tautologous, $PE(G, \mathcal{T}) = \emptyset$ and $MTE(G, \mathcal{T}) = \blacksquare$ because of triviality.*
3. *If G is not tautologous and $\mathcal{F} \models G$, then $PE(G, \mathcal{T}) = \blacksquare$ and $MTE(G, \mathcal{T}) = \emptyset$ because $\blacksquare \not\models G$.*

As a consequence of these properties, the unwarranted conclusion in most AI reasoning that any proposition entails a true proposition ($P \models A \vee \neg A$) is gracefuly encoded as *true* ($\blacksquare$) by the definition of minimal explanation.

So far, the only constraint defined by the relation $\mathcal{R}$ is the subset relation. Different facets of explanations can also be defined by adding more constraint to $\mathcal{R}$. For instance, one useful constraint is to restrict the explanation E to be comprised of only positive assumptions. The obvious application of such definition is in the process of inquiry. Knowing the underlying assumptions that positively support the conclusion is always useful in constructive decision making. Secondly, a *conditional explanation* is defined as $E = Ant \rightarrow Assump$ where $Assump$ is a subset of assumptions and Ant are non-assumptions. The sentence $Ant \rightarrow Assump \rightarrow G$ means that the query G is explainable from $Assump$ subject to the precondition of the facts Ant. These variants of explanations can be defined by extending $\mathcal{R}$ and since the generalization of the relation is non-trial, it deserves a seperate study.

3.3 Extensions

Another type of plausible conclusion is the notion of *irrefutability.* It relies on the concept of an extension of an α-theory which is introduced by the following definition.

Definition 3.4 *Let $\mathcal{T} = (\mathcal{F}, \mathcal{A})$ be an α-theory and for any set of sentences $\mathcal{S}$, define $\Gamma(\mathcal{S})$ to be the smallest set satisfying the following properties.*

1. $\mathcal{F} \subseteq \Gamma(\mathcal{S})$
2. $Th(\Gamma(\mathcal{S})) = \Gamma(\mathcal{S})$
3. *For any formula $\alpha \in \mathcal{A}$, if $\mathcal{S} \cup \alpha$ is consistent then $\alpha \in \Gamma(\mathcal{S})$.*

Thus, a set of sentences $\mathcal{E}$ is an extension of $\mathcal{T}$ if $\Gamma(\mathcal{E}) = \mathcal{E}$.

The next theorem characterizes an extension in a more constructive way following the spirit of Davis [5].

Theorem 3.1 *Let $\mathcal{T} = (\mathcal{F}, \mathcal{A})$ be an α-theory with an enumerable set of assumptions $\mathcal{A}$ and $\phi = \alpha_1, \alpha_2, \ldots$ be some enumeration of $\mathcal{A}$. Define $\mathcal{E}_0^\phi = \mathcal{F}$ and for each i, $i \geq 0$*

$$\mathcal{E}_{i+1}^\phi = \begin{cases} \mathcal{E}_i^\phi \cup \{\alpha_{i+1}\} & \textit{if } \mathcal{E}_i^\phi \cup \{\alpha_{i+1}\} \textit{ is consistent} \\ \mathcal{E}_i^\phi & \textit{otherwise.} \end{cases}$$

Then $\mathcal{E}$ is an extension of $\mathcal{T}$ iff there exists an enumeration ϕ of $\mathcal{A}$ such that $\mathcal{E} = Th(\mathcal{E}_\infty^\phi)$.

Proof : Trivial. □

It follows from the above theorem that an extension is completely determined by the set of assumptions it contains as stated by the following corollary.

Corollary 3.1 *The set $\mathcal{E}$ is an extension of an α-theory $(\mathcal{F}, \mathcal{A})$ iff there is a maximal subset D of $\mathcal{A}$ such that $\mathcal{F} \cup D$ is consistent and $\mathcal{E} = Th(\mathcal{F} \cup D)$.*

Corollary 3.1 implies that for any α-theory $\mathcal{T} = (\mathcal{F}, \mathcal{A})$:

1. If $\mathcal{F}$ is consistent, $\mathcal{T}$ has at least one consistent extension.
2. If $\mathcal{F}$ is inconsistent, the only extension of $\mathcal{T}$ is the whole language $\mathcal{L}$.
3. $\mathcal{T}$ has as many extensions as the number of maximal consistent subsets $\mathcal{A}$ with respect to $\mathcal{F}$.
4. Any subset of $\mathcal{A}$ that is consistent with $\mathcal{F}$ is in some extension of $\mathcal{T}$.

As indicated, any extension of $\mathcal{T}$ is generated by some maximal subset of $\mathcal{A}$ that is consistent with $\mathcal{F}$. Moreover, each one of these subset generates a single extension of $\mathcal{T}$. For this reason, any subset of $\mathcal{A}$ that is maximal and consistent with $\mathcal{F}$ is called an extension generating subset or simply a generating subset of $\mathcal{T}$. Thus, with notions of extension and explanation, a useful question is to ask is whether a sentence is in some known extension. The following lemma establishes the connection between explanation and extension.

Lemma 3.1 *Let $\mathcal{E}$ be an extension of $\mathcal{T} = (\mathcal{F}, \mathcal{A})$ generated by a generating subset $D \subseteq \mathcal{A}$ and G be a sentence. G is in $\mathcal{E}$ iff there is a minimal explanation E of G such that $E \subseteq D$.*

Proof : Assume that E is an explanation of G and D be a maximal consistent subset of $\mathcal{A}$ such that $E \subseteq D$. Since $\mathcal{F} \cup E \models G$, $G \in Th(\mathcal{F} \cup D)$ and G is in the extension $\mathcal{E}$ generated by D. Conversely, if $G \in \mathcal{E}$, $G \in Th(\mathcal{F} \cup D)$ or simply $\mathcal{F} \cup D \models G$. Since $\mathcal{F} \cup D$ is consistent (corollary 3.1) by the definiton of explanation (definition 3.2), D is an explanation of G. By the definition of minimality, there exists an $E \subseteq D$ such that E is a minimal explanation of G. □

As a consequence of the above lemma, we see the relationship between *explainability* and *being in an extension* as expressed by the following corollary.

Corollary 3.2 *A formula G is explainable from $\mathcal{T}$ iff G is in some extension of $\mathcal{T}$.*

Conversely, given an extension $\mathcal{E}$, the task of determining whether a sentence is *not in* this extension $\mathcal{E}$ can also be fomulated, as shown by the following lemma.

Lemma 3.2 *Let $\mathcal{E}$ be an extension of $\mathcal{T} = (\mathcal{F}, \mathcal{A})$ generated by a $D \subseteq \mathcal{A}$ and G be a sentence explainable from $\mathcal{T}$. The sentence G is not in $\mathcal{E}$ iff for every explanation E of G from $\mathcal{T}$, the set $\mathcal{F} \cup E \cup D$ is inconsistent*[5].

Proof : The proof follows from lemma 3.1 as follows: For any explanation $E \subseteq \mathcal{A}$, $E \cup D$ is a set of assumptions consistent with $\mathcal{F}$ iff $E \subseteq D$ simply because D is a maximal subset of assumptions consistent with $\mathcal{F}$. □

[5] i.e. $E \cup D$ is an inconsistent subset of assumptions.

Finally, an *irrefutable* sentence is naturally defined to be a sentence that is in all extensions of $\mathcal{T}$. Equally important, an *irrefutable* sentence G implies that for every extension, there is a consistent explanation of G with respect to $\mathcal{T}$.

Definition 3.5 *Given an α-theory $\mathcal{T} = (\mathcal{F}, \mathcal{A})$ and a sentence G. The sentence G is irrefutable in $\mathcal{T}$ if G is in every extension of $\mathcal{T}$.*

Clearly, a formula G is irrefutable in $\mathcal{T}$ if for any extension $\mathcal{E}$ of $\mathcal{T}$ generated by some D, there is a minimal explanation E of G such that $E \subseteq D$. This implies that to examine irrefutability using definition 3.5 would require generating all the extensions of $\mathcal{T}$. An alternative characterization of an irrefutable sentence, which does not require explicit reference to the extensions of $\mathcal{T}$ is given by the following theorem.

Theorem 3.2 (Irrefutable) *Let G be a sentence explainable from $\mathcal{T} = (\mathcal{F}, \mathcal{A})$ such that the set of its minimal explanations from $\mathcal{T}$ is finite; that is $ME(G, \mathcal{T}) = \{E_1, E_2, \ldots, E_k\}$. The sentence G is in every extension of $\mathcal{T}$ iff $\neg E_1 \wedge \ldots \wedge \neg E_k$ is not explainable from $\mathcal{T}$.*

Proof : If: Assume there exists an extension $\mathcal{E}$ of $\mathcal{T}$ generated by D, the sentence G is not in $\mathcal{E}$ and $\neg E_1 \wedge \ldots \wedge \neg E_k$ is not explainable from $\mathcal{T}$. By lemma 3.2, $\mathcal{F} \cup E_i \cup D$ is inconsistent for each i, $1 \leq i \leq k$. Since $\mathcal{F} \cup D$ is consistent by corollary 3.1 and by propositional reasoning $\mathcal{F} \cup D \models \neg E_i$ for each i, $1 \leq i \leq k$. Thus, $\mathcal{F} \cup D \models \neg E_1 \wedge \ldots \wedge \neg E_k$ or simply $\mathcal{F} \models D \rightarrow \neg E_1 \wedge \ldots \wedge \neg E_k$. Consequently, D is an explanation of $\neg E_1 \wedge \ldots \wedge \neg E_k$ from $\mathcal{T}$ which contradicts the assumption that it is not explainable from $\mathcal{T}$.

Only if: Assume that G is in every extension and $\neg E_1 \wedge \ldots \wedge \neg E_k$ is explainable from $\mathcal{T}$. Since $\neg E_1 \wedge \ldots \wedge \neg E_k$ is explainable from $\mathcal{T}$, there exists an explanation $E \subseteq D$ and an extension $\mathcal{E}$ generated by D such that $\mathcal{F} \cup D \models \neg E_1 \wedge \ldots \wedge \neg E_k$ by lemma 3.1. Hence by propositional reasoning, for every i, $1 \leq i \leq k$, $\mathcal{F} \cup E_i \cup D$ is inconsistent. Consequently by lemma 3.2, G is not in $\mathcal{E}$ contradicting the assumption that G is in every extension. □

The concept of a logical extension generated by a set of assumptions is interesting in many aspects. For instance, the consistency of a subset of assumptions can be determined by comparing it to the collection of extensions. In short, the characterization of the assumption based abduction and extension presented here has provided more expressiveness to reasoning systems. Even more interesting is that the computations for these features will be defined using the homogeneous representation of the $ACMS$ (more on section 4).

3.4 Direct Consequences

In AI reasoning, most often the type of consequence we desire is precisely the consequence that is most related and relevant to the query. We can view this as a kind of logical focus of attention. For instance, given some facts $\mathcal{F}$ and a query G, it is desirable to know whether a sentence C is a consequence of $\mathcal{F} \cup G$ but is not a consequence from $\mathcal{F}$ alone. By augmenting this notion of consequence with assumptions, one can view this process as an inquiry into which assumption follows from the fact G with respect to $\mathcal{F}$. We shall call this the *assumption based deductive reasoning.*

Definition 3.6 *Let $\mathcal{T} = (\mathcal{F}, \mathcal{A})$ be an α-theory and the query G be a sentence. A set of sentences C in disjunction (that is, the sentence $\vee C$) is a direct consequence*[6] *of G with respect to $\mathcal{T}$ if*

1. $C \subseteq \mathcal{A} \cup \overline{\mathcal{A}}$,
2. $\mathcal{F} \cup G \models C$ *and*
3. $\mathcal{F} \not\models C$.

Although in principal direct consequences are more closely related to G itself than to $\mathcal{F}$, the number of direct consequences of a single formula G can be very large or even infinite. Likes its counter-part *explanation*, we shall introduce a kind of minimality restriction.

Definition 3.7 *Let $\mathcal{T} = (\mathcal{F}, \mathcal{A})$ be an α-theory , the query G be a sentence and a direct consequence C of G with respect to $\mathcal{T}$.*

1. *C is a minimal direct consequence of G if C is a direct consequence of G and there is no other direct consequence C' of G such that C' subsumes C.*
2. *C is a trivial direct consequence of G if $G \models C$ otherwise it is non-trivial.*
3. *C is a minimal trivial direct consequence of G if C is both minimal and trivial.*

[6] Hereon, it is understood that a direct consequence is a set of sentences C in disjunction, that is the sentence $\vee C$.

4. C is a prime direct consequence of G if it is both minimal and non-trivial.

For convenience, we will use $MDC(G,\mathcal{T})$, $MTDC(G,\mathcal{T})$ and $PDC(G,\mathcal{T})$ to denote respectively, the sets of minimal, minimal trivial and prime direct consequences of G with respect to $\mathcal{T}$. It trivially follows from the definitions that the property $MDC(G,\mathcal{T}) = PDC(G,\mathcal{T}) \cup MTDC(G,\mathcal{T})$ holds. Note also that, assuming a consistent set of $\mathcal{F}$, these minimalities also have the following properties:

1. If $\mathcal{F} \cup G$ is inconsistent, then $MDC(G,\mathcal{T}) = \square$. This is obvious since $\mathcal{F} \cup G \models \square$.
2. If G is a tautology, then $PDC(G,\mathcal{T}) = \emptyset$ and $MTDC(G,\mathcal{T}) = \square$. Due to triviality, the *smallest* direct consequence for G is $\square$ (false).
3. If G is not a tautology and $\mathcal{F} \models G$, then $MDC(G,\mathcal{T}) = \emptyset$. It follows from the definition that there is no sentence C can be a direct consequence because when $\mathcal{F} \models G \rightarrow C$, $\mathcal{F} \models C$.

Note that the notion of a minimal direct consequence provides an answer to the problem of superfluous consequences that is often critized by the AI reasoning community. That is, if $\mathcal{F} \cup G$ is inconsistent, the only conclusion (minimal direct consequence) is the empty sentence (false) as opposed to concluding any sentences in classical deduction. Similarly, if $\mathcal{F} \models G$, the sentence G is the conclusion (minimal direct consequence) and no other conclusion disjunctively attached to G is accepted. For instance, if the facts $\mathcal{F}$ prove that *"salt is soluble in water"*, we cannot conclude a minimal direct consequence that *"salt is soluble in water"* or *"Unicorns exist"*.

Two refined notions of the minimal direct consequence are especially advantageous in some application domains. The first refinement is obtained by restricting C to contain only negative assumptions (i.e. $C = \{\overline{C_1}, \ldots, \overline{C_n}\}$, for each C_i appears positive in $\mathcal{A}$). In this case, each C_i represents an assumption such that the conjunction of all of them is in conflict with G that is, $\mathcal{F} \cup G \cup \overline{C}$ is inconsistent. This is particularly useful in identifying potential conflicts among assumptions. For instance, if C is such a direct consequence for G, then the extension generating subset D of $\mathcal{T}$ that is a superset of $\overline{C}$ will be split for the new theory with $\mathcal{F} \cup G$.

Secondly, C is restricted to the form $(Assump \rightarrow Conseq)$ where $Assump$ is a conjunction of assumptions while no assumption occurs in $Conseq$. Thus $G \rightarrow (Assump \rightarrow Conseq)$ states that $Conseq$ is subjectively entailed by G with respect to the assumptions $Assump$. We shall call this the *conditional direct consequence* of G.

Obviously many forms of minimal direct consequences can be defined depending on the application. The computation for these varieties of consequences will be achieved in the $ACMS$ (more on section 4).

4 An Assumption-Based Clause Management System

In the previous section, we have discussed the functionality of ***assumption based reasoning*** in terms of *direct consequence*, *explanation*, *agreement*, *irrefutability* and *extension*. In this section we define an ***Assumption Based Clause Management System*** ($ACMS$) that performs this type of reasoning. In [17] we studied a *Clause Management System* (CMS) that computes the set of minimal supports for any sentence in CNF. Since direct consequences, explanations and supports share some common properties, the $ACMS$ is expected to be an extension to the CMS that takes into account the set of assumptions and their usage.

4.1 Restricted α-theory

Prior to the discussion, we shall restrict the theory in a more computationally realistic realm. So far, we have dealt with general α-theory $\mathcal{T} = (\mathcal{F}, \mathcal{A})$ with potentially infinite sets of facts and assumptions. For pragmatic reasons and computational feasibility, we now restrict the sets $\mathcal{F}$ and $\mathcal{A}$ of $\mathcal{T}$ to be finite. Without lost of generality we shall also assume $fact$ is a set of sentences in CNF. Lastly, we shall impose a restriction that an assumption is to be a single literal. The latter restriction is justified by the following discussion.

Definition 4.1 (σ-transformation) *Let $\mathcal{T} = (\mathcal{F}, \mathcal{A})$ be an α-theory . We define a transformation σ as follows.*

1. *For every sentence α in $\mathcal{A}$, $\sigma(\alpha) = \alpha$ if α is a single propositional literal. Otherwise $\sigma(\alpha) = \lambda$ where λ is a new propositional variable not used anywhere in the theory.*
2. $\sigma(\mathcal{A}) = \{\sigma(\alpha) \mid \alpha \in \mathcal{A}\}$
3. $\sigma(\mathcal{F}) = \mathcal{F} \cup \{\sigma(\alpha) \equiv \alpha \mid \alpha \in \mathcal{A}$ *and* $\sigma(\alpha) \neq \alpha\}$ *and*
4. $\sigma(\mathcal{T}) = (\sigma(\mathcal{F}), \sigma(\mathcal{A}))$.

The intuition behind the σ-transformation is that if the sentence α in $\mathcal{A}$ is not a single literal, it is replaced by some new variable λ and a new sentence expressing the equivalence of α

and λ is added in $\mathcal{F}$. If σ-transformation is performed on both $\mathcal{F}$ and $\mathcal{A}$, then obviously $\mathcal{T}$ and $\sigma(\mathcal{T})$ are equivalent as expressed in the following theorem.

Theorem 4.1 *For any α-theory $\mathcal{T} = (\mathcal{F}, \mathcal{A})$, $\mathcal{T}$ and $\sigma(\mathcal{T})$ are equivalent in the sense that for any sentence G, E is an explanation/direct consequence of G from $\mathcal{T}$ iff $\sigma(E)$ is an explanation/direct consequence of G from $\sigma(\mathcal{T})$.*

Proof : Trivially follows from definition 4.1 and propositional reasoning. □

Hereafter, we can safely assume that for any α-theory $\mathcal{T} = (\mathcal{F}, \mathcal{A})$, $\mathcal{F}$ is a finite set of sentences in CNF and the finite set of assumptions $\mathcal{A}$ contains only single literals as elements.

4.2 Explanations, Direct Consequences and Supports

With the restriction on CNF enforced, a careful examimation of the definition of explanation suggests that an explanation of G is the negation of an assumption based support for G with respect $\mathcal{T}$, and is stated more formally in the following lemma.

Lemma 4.1 *Let $\mathcal{T} = (\mathcal{F}, \mathcal{A})$ be an α-theory and G be a sentence. A conjunctive clause E is an explanation of G iff $E \subseteq \mathcal{A}$ and $\neg E$ is a support for G with respect to $\mathcal{T}$.*

Proof : With respect to $\mathcal{T}$, the conjunctive clause E is an explanation of G iff $E \subseteq \mathcal{A}$, $\mathcal{F} \cup E \models G$ and $\mathcal{F} \cup E$ is consistent (by definition 3.2). Propositionally this is equivalent to $\mathcal{F} \models \neg E \vee G$ and $\mathcal{F} \not\models \neg E$. By the definition of support, $\neg E$ is a support for G. □

Minimality, triviality and primeness are defined in similar fashion. Finally, the next lemma reveals the connection between direct consequence and assumption based support.

Lemma 4.2 *Let $\mathcal{T} = (\mathcal{F}, \mathcal{A})$ be an α-theory and G be a sentence. A disjunctive clause C is a direct consequence of G iff $C \subseteq \mathcal{A} \cup \overline{\mathcal{A}}$ and C is a support for $\neg G$ with respect to $\mathcal{T}$.*

Proof : It follows from the definition of direct consequence (3.6) and support. □

Similarly, minimal, minimal trivial and prime direct consequence can be shown to correspond to supports in their appropriate form. Lemmata 4.1 and 4.2 indicate that the computational effort for finding minimal explanations and direct consequences of a sentence G, is tantamount to searching for the assumption based minimal supports of G. In addition, the computation of irrefutability and agreement can also be reduced to finding assumption based minimal supports. Consequently, we need an extension of the CMS that will efficiently compute assumption based minimal (as well as prime and minimal trivial) supports.

4.3 Computations

Given an α-theory $\mathcal{T} = (\mathcal{F}, \mathcal{A})$, the $ACMS$ represents the α-theory by the following sets of sentences. The set of sentences $\mathcal{F}$ in CNF is represented by the set of its prime implicates $PI(\mathcal{F})$ in CNF, and the set of assumptions $\mathcal{A}$ is represented by a set of literals. In the event of an update, that is when adding a new disjunctive clause in $\mathcal{F}$, the set of prime implicates of the new facts are updated using an incremental algorithm for computing prime implicates which was studied in [10]. In the event of a query, the computation of the various types of responses to the query is achieved by the following methods. Firstly, given an α-theory $\mathcal{T} = (\mathcal{F}, \mathcal{A})$ and a sentence G, we shall identify the various types of responses as follows:

(i) computing the minimal, minimal trivial or prime explanations of G;
(ii) asserting whether G is agreeable or irrefutable;
(iii) computing the extensions of $\mathcal{T}$; and
(iv) computing the minimal, minimal trivial or prime direct consequences of G.

We shall begin by examining the computation of (i) and (ii). Since all definitions presented here are merely extensions of the definitions of support, the correctness of the methods used hereafter trivially follows from the corresponding algorithms for computing supports discussed in [15, 17]. The set of prime explanations for a disjunctive clause G is computed as:

$$PE(G, \mathcal{T}) = SUB(\{\overline{S} \mid S = P - G, P \in PI(\mathcal{F}), P \cap G \neq \emptyset, P \cup G \text{ is fundamental and } \overline{S} \subseteq \mathcal{A}\}).$$

Similarly, the set of minimal trivial explanations for a disjunctive clause G is computed as:

1. if $\mathcal{F}$ is consistent and G is non-fundamental, $MTE(G, \mathcal{T}) = \{\Box\}$, otherwise
2. if $PE(G, \mathcal{T}) = \{\Box\}$ then $MTE(G, \mathcal{T}) = \emptyset$, otherwise
3. $MTE(G, \mathcal{T}) = \{e \mid e \in G, e \in \mathcal{A} \text{ and } \overline{e} \notin PI(\mathcal{F})\}$.

Consequently, the set of minimal explanations for a disjunctive clause G is computed by taking the union of the above two sets, that is $ME(G,\mathcal{T}) = PE(G,\mathcal{T}) \cup MTE(G,\mathcal{T})$.

On the other hand, if $G = \{G_1, \ldots, G_n\}$ is a set of sentences in CNF, the set of minimal explanations of G is computed recursively by[7]

$$\begin{aligned} ME(G,\mathcal{T}) = SUB(\{\ & E \wedge E' \mid E \in ME(G_1 \wedge \ldots \wedge G_{n-1}, \mathcal{T}) \text{ and} \\ & E' \in ME(G_n, \mathcal{T}) \text{ and} \\ & \text{no } P \in PI(\mathcal{F}) \text{ subsumes } \overline{E} \vee \overline{E'} \text{ and} \\ & \overline{E} \vee \overline{E'} \text{ is fundamental } \}). \end{aligned}$$

Putting together these methods, an algorithm for computing the set of minimal explanations for a CNF sentence G can be stated as follows:

Algorithm for Minimal Explanation
Input: $PI(\mathcal{F})$ and a sentence G both in CNF.
Output: $ME(G,\mathcal{T}) = PE(G,\mathcal{T}) \cup MTE(G,\mathcal{T})$.

```
Step 1.0 If $PI(\mathcal{F}) = \{\Box\}$ then $MTE(G,\mathcal{T}) = \emptyset$ and $PE(G,\mathcal{T}) = \emptyset$, GOTO 6.0.
Step 2.0 If $G$ is non-fundamental then $PE(G,\mathcal{T}) = \emptyset$ and $MTE(G,\mathcal{T}) = \{\Box\}$, GOTO 6.0.
Step 3.0 $PE(G,\mathcal{T}) = SUB(\{\overline{S} \mid S = P - G, P \in PI(\mathcal{F}), P \cap G \neq \emptyset, P \cup G$ is fundamental and $\overline{S} \subseteq \mathcal{A}\})$
Step 4.0 If $PE(G,\mathcal{T}) = \{\Box\}$ then $MTE(G,\mathcal{T}) = \emptyset$, GOTO 6.0.
Step 5.0 $MTE(G,\mathcal{T}) = \{e \mid e \in G \text{ and } \overline{e} \notin PI(\mathcal{F})\}$
Step 6.0 RETURN: $ME(G,\mathcal{T}) = PE(G,\mathcal{T}) \cup MTE(G,\mathcal{T})$.
```

Having presented an algorithm for computing minimal explanations, the next discussion is on the computation of *agreement* and *irrefutability*. A method for asserting agreement is suggested by the following lemma.

Lemma 4.3 *Given an α-theory $\mathcal{T} = (\mathcal{F}, \mathcal{A})$ and a CNF sentence G, G is agreeable with respect to $\mathcal{T}$ if $ME(G,\mathcal{T}) \neq \emptyset$ and $ME(\overline{G},\mathcal{T}) = \emptyset$.*

Proof : Follows from definition 3.2. □

We can also effectively decide the irrefutability of a CNF sentence G in $\mathcal{T}$ by using the result of theorem 3.2. More precisely, we first compute the set of minimal explanations

[7] The correctness proof for the minimal support for a CNF sentence can be found in [17].

$\Phi = ME(G, \mathcal{T})$ and then using the same method, compute the set of minimal explanations for the conjunction of the negations of all explanations in Φ, that is,

$$\Phi' = ME(\bigwedge_{S \in \Phi} \overline{S}, \mathcal{T}).$$

If the set $\Phi \neq \emptyset$ and the set $\Phi' = \emptyset$, then G is irrefutable in $\mathcal{T}$ otherwsie it is not irrefutable. The reader should note that minimal supports (instead of minimal explanations) are enough to determine irrefutability because of the duality. More precisely, the set of minimal supports $\Sigma = MS(G, \mathcal{T})$ and

$$\Sigma' = MS(\bigwedge_{S \in \Sigma} S, \mathcal{T})$$

need to be computed instead of Φ and Φ'. If $\Sigma \neq \emptyset$ and $\Sigma' = \emptyset$, G is irrefutable in $\mathcal{T}$ and it is not otherwise.

Even though we have a method to deduce whether a sentence is agreeable or irrefutable, that is whether it is in some extension or is in every extension, without explicitly computing all the extensions of an α-theory , there are cases where computing all the extensions or the set of extension generating subsets is also desirable. For instance, in a *Reasoner-ACMS* framework, the *Reasoner* can query the *ACMS* for some or all maximal consistent subsets of assumptions[8], that is the set of extension generating subsets, with respect to the current environment $(\mathcal{F}, \mathcal{A})$.

It is important to note that any α-theory $\mathcal{T} = (\mathcal{F}, \mathcal{A})$ with a finite set of assumptions has finitely many extensions. Since each extension of an α-theory is completely characterized by its generating subsets, computing all the extensions of the theory is equivalent to computing all its generating subsets. The first observation is that the set $PI(\mathcal{F})$ gives us for free the set of *some* minimal inconsistent subsets of assumptions, which Reiter [14] called minimal conflict sets and de Kleer [6] refered to as nogoods. For reason of coherency we shall follow Reiter's terminology.

Definition 4.2 (Conflict Sets) *Given an α-theory $\mathcal{T} = (\mathcal{F}, \mathcal{A})$, $C \subseteq \mathcal{A}$ is a minimal conflict set of $\mathcal{T}$ if $\mathcal{F} \cup C$ is inconsistent and no proper subset of it is inconsistent with $\mathcal{F}$.*

The following lemma explicitly characterizes conflict sets in terms of the minimal implicates of $\mathcal{T}$, and the subsequent corollary describes conflict sets in terms of prime implicates.

[8] In de Kleer's terminology, it is call the maximal consistent environments [6].

Lemma 4.4 *Given a set $C = \{c_1, \ldots, c_k\}$ where $C \subseteq \mathcal{A}$, C is a minimal conflict set of the α-theory $\mathcal{T} = (\mathcal{F}, \mathcal{A})$ iff $MI(\mathcal{F}) \models \neg c_1 \vee \ldots \vee \neg c_k$ or equivalently, the set $\{\neg c_1, \ldots, \neg c_k\} \in MI(\mathcal{F})$.*

Proof : Follows from definition 4.2 and the entailment property of minimal implicates. □

Intuitively, if $\mathcal{F} \cup C$ is inconsistent then $\mathcal{F} \models \overline{C}$ and $MI(\mathcal{F}) \models \overline{C}$. Using the entailment property of minimal implicates, there is a minimal implicate $P \in MI(\mathcal{F})$ that subsumes $\overline{C}$. Hence P is a *minimal conflict set* of $\mathcal{T}$. Since the set $MI(\mathcal{F})$ can be constructed from $PI(\mathcal{F})$, the following corollary describes explicitly the computation for generating minimal conflict sets using prime implicates of $\mathcal{F}$.

Corollary 4.1 *Given $C = \{c_1, \ldots, c_k\}$ where $C \subseteq \mathcal{A}$, C is a minimal conflict set of the α-theory $\mathcal{T} = (\mathcal{F}, \mathcal{A})$ iff $\{\neg c_1, \ldots, \neg c_k\} \in \{PI(\mathcal{F}) \cup \{\{x, \neg x\} \mid x, \neg x \in \mathcal{A}$ and $\{x\}, \{\neg x\} \notin PI(\mathcal{F})\}\}$.*

Proof : It follows from lemma 4.4 and theorems 2.1 and 2.2 of [17] that $MI(\mathcal{F}) = PI(\mathcal{F}) \cup MTI(\mathcal{F})$ and $MTI(\mathcal{F}) = \{\{x, \neg x\} \mid x \in \mathcal{V}$ and no $P \in PI(\mathcal{F})$ subsumes $\{x, \neg x\}\}$ where $\mathcal{V}$ is the language vocabulary. □

For convenience, we shall denote the set of minimal conflict sets for the α-theory as $MCS(\mathcal{F}, \mathcal{A})$. Also note that it can be subdivided into two disjoint sets, namely the *prime conflict set* $PCS(\mathcal{F}, \mathcal{A})$ corresponding to elements derived from $PI(\mathcal{F})$, and the other derived from $MTI(\mathcal{F})$ is the *minimal trivial conflict set* $MTCS(\mathcal{F}, \mathcal{A})$. Thus the minimal conflict set is the union of the prime conflict set and the minimal trivial conflict set. Note that the set $MTCS(\mathcal{F}, \mathcal{A})$ is necessary by virtue of the definition of conflict set 4.2 and lemma 4.4. For instance, if the literals z, $-z \in \mathcal{A}$ and neither of them occur in any form in $\mathcal{F}$, then obviously $\mathcal{F} \cup \{z, -z\}$ is inconsistent. This suggests that $\{z, -z\}$ is a minimal conflict set and $PI(\mathcal{F})$ alone does not produce this minimal conflict set.

Having extracted the minimal conflict sets from the sets of prime implicates and minimal trivial implicates of $\mathcal{F}$, we can compute their minimal hitting sets as defined by Reiter [14] which will eventually lead to our notion of extension.

Definition 4.3 (Reiter's Hitting Sets) *Suppose $\mathcal{W}$ is a collection of subsets of $\mathcal{A}$, a set $H \subseteq \mathcal{A}$ is a hitting set for $\mathcal{W}$ if $H \cap C \neq \emptyset$ for each $C \in \mathcal{W}$. A hitting set H for $\mathcal{W}$ is minimal iff no proper subset of H is a hitting set for $\mathcal{W}$.*

Intuitively, a hitting set is a set that has elements in common with every set in $\mathcal{W}$. For instance, if $\mathcal{W} = \{\{1,2\},\{3,4\},\{5,6\}\}$, then the set $\{1,3,5\}$ is a hitting set of $\mathcal{W}$. The extension generating subsets of $\mathcal{T}$ are exactly the complement of the minimal hitting sets of $MCS(\mathcal{F},\mathcal{A})$. The following theorem characterizes extensions in terms of minimal hitting sets.

Theorem 4.2 *Let $\mathcal{T} = (\mathcal{F},\mathcal{A})$ be an α-theory , $\mathcal{W} = \{C_1,\ldots,C_k\}$ is the set of all minimal conflict sets of $\mathcal{T}$ and $\mathcal{H}$ is the set of all minimal hitting sets of $\mathcal{W}$. A subset D of $\mathcal{A}$ is an extension generating subset iff the set $\mathcal{A} - D \in \mathcal{H}$.*

Proof : *If:* Assume that $\mathcal{A} - D \in \mathcal{H}$, then by definition 4.3, for each i, $1 \le i \le k$, $(\mathcal{A} - D) \cap C_i \neq \emptyset$. Consequently for all i, $C_i \not\subset D$ for otherwise $(\mathcal{A} - D) \cap C_i = \emptyset$. Since every C_i is a minimal conflict set and every $C_i \not\subset D$, $\mathcal{F} \cup D$ is consistent and D is in some extension generating subset. Suppose D is not maximal i.e. there is an $a \in \mathcal{A} - D$ such that $\mathcal{F} \cup D \cup \{a\}$ is consistent. Then obviously $C_i \not\subset D \cup \{a\}$ for any i, $1 \le i \le k$ for otherwise $\mathcal{F} \cup D \cup \{a\}$ is inconsistent. Hence $(\mathcal{A} - (D \cup \{a\})) \cap C_i \neq \emptyset$ for all i, $1 \le i \le k$ and by definition 4.3, $\mathcal{A} - (D \cup \{a\})$ is a hitting set. But then $(\mathcal{A} - (D \cup \{a\})) \subset \mathcal{A} - D$, which contradicts that $\mathcal{A} - D \in \mathcal{H}$.

Only if: Assume that a subset D of $\mathcal{A}$ is an extension generating subset. Then by corollary 3.1, D is a maximal subset of $\mathcal{A}$ such that $\mathcal{F} \cup D$ is consistent. Hence D is a maximal subset such that each $C_i \not\subseteq D$, for each i, $1 \le i \le k$. Therefore $\mathcal{A} - D$ is a minimal subset such that $(\mathcal{A} - D) \cap C_i \neq \emptyset$ for each i, $1 \le i \le k$ and by definition 4.3, $\mathcal{A} - D \in \mathcal{H}$. □

Since all the minimal conflict sets are readily available, the method for computing hitting sets is simplified. Let each minimal conflict set C_i be a disjunctive clause, $1 \le i \le k$ and their conjunction $\wedge_{i=1}^{k} C_i$ is a sentence in CNF. Let $\vee_{i=1}^{m} H_i$ be the sentence obtained after transforming $\wedge_{i=1}^{k} C_i$ into DNF and simplifying it by deleting subsumed clauses. Then the sets H_i, $1 \le i \le m$ are all and only the minimal hitting sets for the minimal conflict sets. Additionally, there is an extra constraint on the minimal hitting set as characterized by the following lemma.

Lemma 4.5 *Let $\mathcal{T} = (\mathcal{F},\mathcal{A})$ be an α-theory and $\mathcal{C}$ be the set of all minimal conflict sets of $\mathcal{T}$. No minimal hitting set H of $\mathcal{C}$ contains complementary literals.*

Proof : Assume $\mathcal{F}$ is consistent and hence $PI(\mathcal{F})$ does not contain the empty clause. Let $\mathcal{C}$ be the set of all minimal conflict sets of $\mathcal{T}$, and $\mathcal{H}$ be the set of all minimal hitting sets of $\mathcal{C}$. Suppose

$L \in \mathcal{H}$ and the complimentary literals $a, \overline{a} \in L$:

(a) First we prove that neither $\{a\}$ nor $\{\neg a\}$ are in $PI(\mathcal{F})$. Assume otherwise, i.e. $\{a\} \in PI(\mathcal{F})$, then "$\neg a$" cannot occur in any of the clauses $C \in PI(\mathcal{F})$ because the resolvent of C and "a" would subsume C, contradicting C being a prime implicate of $\mathcal{F}$. Therefore, by the definition of conflict set 4.2, $\{\neg a\} \in MCS(\mathcal{F}, \mathcal{A})$ and subsequently, the set $\{a, \neg a\} \notin MCS(\mathcal{F}, \mathcal{A})$. Consequently, since only $\{\neg a\} \in MCS(\mathcal{F}, \mathcal{A})$, and the literal "$a$" does not occur in any conflict set, therefore by the definition of minimal hitting set 4.3, $a \notin L$ contradicting the assumption. A similar argument holds for $\{\neg a\} \in PI(\mathcal{F})$, therefore, $\{a\}$, $\{\neg a\} \notin PI(\mathcal{F})$.

(b) Now we will show that L is not a minimal hitting set. Assume otherwise, then by theorem 4.2, the set $\mathcal{A} - L$ is a maximal consistent subset of assumptions with respect to $\mathcal{F}$. But neither $\{a\}$ nor $\{\neg a\}$ is in $PI(\mathcal{F})$ which implies that $\mathcal{F} \cup \{a\}$ is consistent and $\mathcal{F} \cup \{\neg a\}$ is also consistent respectively. Consequently, $(\mathcal{A} - L) \cup \{a\}$ and $(\mathcal{A} - L) \cup \{\neg a\}$ are consistent with $\mathcal{F}$ and they both contain $\mathcal{A} - L$ contradicting the maximality of $\mathcal{A} - L$. □

Clearly any fast method for transforming from CNF to DNF is suitable for our purpose. For example, we can represent the CNF formula as a matrix and use the connection method [3] to construct a set of paths $\mathcal{P}$ through the matrix. It can be shown easily that the DNF formula is the set of non-complementary paths $\mathcal{P}'$ that are not subsumed by other paths in the set $\mathcal{P}$. Furthermore, such techniques can be optimized for our setting since subsumptions can be greatly reduced by examining the structure of the matrix. More formally, let $\mathcal{M}$ be a set of sets represented by a matrix where each $M_i \in \mathcal{M}, 1 \leq i \leq n$ is a column. A *path* is defined as a set $\{m_i \mid m_i \in M_i, \text{ for all } i = 1, n\}$. For example, let $\mathcal{M} = \{\{a, b, c\}, \{a, \neg c\}, \{f, g\}\}$ and its corresponding matrix is:

$$\begin{pmatrix} a & a & f \\ b & \neg c & g \\ c & & \end{pmatrix}$$

A possible path in $\mathcal{M}$ is the set $\{a, \neg c, f\}$. Using the definition of a path above, we can define a simple recursive procedure to enumerate a path as follows:

$$Path(\mathcal{M}) = \begin{cases} 1. \quad \{m \mid m \in \mathcal{M}\} & \text{if } \mathcal{M} \text{ is a clause,} \\ 2. \quad Path(M_1, \ldots, M_{n-1}) \cup Path(M_n) & \text{if } \mathcal{M} = M_1, \ldots, M_n. \end{cases}$$

For optimization, additional constraints can be incorporated into the process of selecting a literal in statement (1). That is, a literal "m" is selected from M_i in $Path(M_i)$ if $\neg m \notin Path(M_1, \ldots, M_{i-1})$. Conversely, if $Path(M_1, \ldots, M_{i-1}) \cap M_i \neq \emptyset$, then the whole column M_i can be ignored *with respect to* $Path(M_1, \ldots, M_{i-1})$ without loss of completeness because of minimality. Note that this is not true with respect to other paths. The reader can examine the validity of the claim by trying out the matrix shown above.

Finally, the last service the $ACMS$ is designed to provide is the computation of direct consequences. Note that by lemma 4.2, a direct consequence of a sentence G corresponds to a support for the negation of G. Obviously, computing direct consequences is equivalent to computing supports, a function already performed by the $ACMS$. Conditional explanations and direct consequences can be easily computed by extending lemma 4.1 and 4.2 to include conditional. The definition of negative literals assumption as direct consequence of G is computed by restricting the direct consequence of G to consists of solely negative literals. The varieties of explanations and direct consequences are enormous and their significance will definitely be dependent on the context in which question and answer are formulated. Nevertheless, most of them are expected to be treated by the $ACMS$ in a way not too different to that discussed here.

5 Example

To illustrate the concepts introduced thus far, let us consider an application in Boolean circuit diagnosis. The notion of diagnosis presented here covers more than the conventional notion of fault diagnosis. It includes the notion of inquiry into the system behaviour in both normal and faulty state. The responce to a query can be of an explanation, a conditional explanation, a set of extensions, a direct consequence and many others. The varieties of question answering provided by the $ACMS$ in this domain will be the focus of this exercise.

Consider the following five gate full adder (figure 1). The gates X_1 and X_2 are *xor-gates*; A_1 and A_2 are *and-gates*; and O_1 is an *or-gate*. The task here is to investigate the system behavior given the system description (SD) of the circuit and a set of observed binary inputs and outputs (OBS). Descriptions of the system behaviors express the relationship between an OBS and the normality or abnormality of the components encoded as assumptions. The normality and abnormality of a component X_1 are expressed as $\neg ab(X_1)$ and $ab(X_1)$ respectively. Thus,

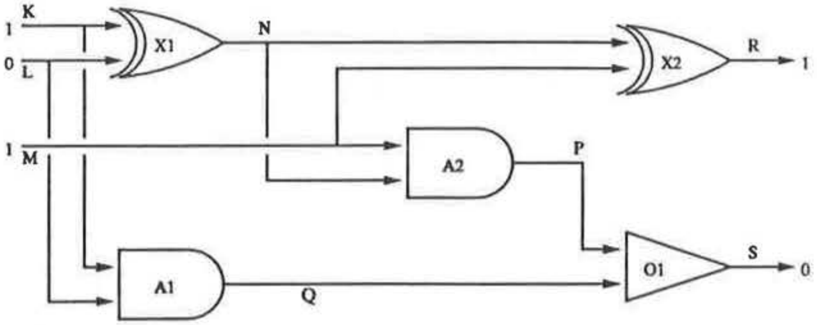

Figure 1: A 1-bit Full Adder

the complete set of assumptions for SD is the set

$$\mathcal{A} = \{ab(X_1), ab(X_2), ab(A_1), ab(A_2), ab(O_1), \neg ab(X_1), \neg ab(X_2), \neg ab(A_1), \neg ab(A_2), \neg ab(O_1)\}.$$

For all the queries from here on, if the set of assumptions is not explicitly stated, this will be the intended set of assumptions. The system description of a component is coded in the form $effects \rightarrow causes$ or sometime $causes \rightarrow effects$ where *causes* are assumptions of normality and abnormality; and the *effects* are the boolean values the wires contain. We shall also assume equality and inequality axioms in SD which are not explicitly stated. The following set SD defines both the normal component specification, e.g. $K = 0 \wedge L = 0 \wedge N = 0 \rightarrow \neg ab(X_1)$; and the abnormal component specification, e.g. $K = 0 \wedge L = 0 \wedge N = 1 \rightarrow ab(X_1)$.

$$SD = \{ \qquad \textbf{Normal Component Specification}$$

$$\begin{array}{ll}
K = 0 \wedge L = 0 \wedge N = 0 \rightarrow \neg ab(X_1), & N = 0 \wedge M = 0 \wedge R = 0 \rightarrow \neg ab(X_2), \\
K = 0 \wedge L = 1 \wedge N = 1 \rightarrow \neg ab(X_1), & N = 0 \wedge M = 1 \wedge R = 1 \rightarrow \neg ab(X_2), \\
K = 1 \wedge L = 0 \wedge N = 1 \rightarrow \neg ab(X_1), & N = 1 \wedge M = 0 \wedge R = 1 \rightarrow \neg ab(X_2), \\
K = 1 \wedge L = 1 \wedge N = 0 \rightarrow \neg ab(X_1), & N = 1 \wedge M = 1 \wedge R = 0 \rightarrow \neg ab(X_2), \\
 & \\
K = 1 \wedge L = 1 \wedge Q = 1 \rightarrow \neg ab(A_1), & M = 1 \wedge N = 1 \wedge P = 1 \rightarrow \neg ab(A_2), \\
K = 0 \wedge \qquad\quad Q = 0 \rightarrow \neg ab(A_1), & M = 0 \wedge \qquad\quad P = 0 \rightarrow \neg ab(A_2), \\
\qquad\quad L = 0 \wedge Q = 0 \rightarrow \neg ab(A_1), & \qquad\quad N = 0 \wedge P = 0 \rightarrow \neg ab(A_2), \\
 & \\
P = 0 \wedge Q = 0 \wedge S = 0 \rightarrow \neg ab(O_1), & \\
P = 1 \wedge \qquad\quad S = 1 \rightarrow \neg ab(O_1), & \\
\qquad\quad Q = 1 \wedge S = 1 \rightarrow \neg ab(O_1), &
\end{array}$$

$$\textbf{Abnormal Component Specification}$$

$$\begin{array}{ll}
K = 0 \wedge L = 0 \wedge N = 1 \rightarrow ab(X_1), & N = 0 \wedge M = 0 \wedge R = 1 \rightarrow ab(X_2), \\
K = 0 \wedge L = 1 \wedge N = 0 \rightarrow ab(X_1), & N = 0 \wedge M = 1 \wedge R = 0 \rightarrow ab(X_2), \\
K = 1 \wedge L = 0 \wedge N = 0 \rightarrow ab(X_1), & N = 1 \wedge M = 0 \wedge R = 0 \rightarrow ab(X_2),
\end{array}$$

$$
\begin{array}{ll}
K=1 \wedge L=1 \wedge N=1 \rightarrow ab(X_1), & N=1 \wedge M=1 \wedge R=1 \rightarrow ab(X_2), \\[1em]
K=1 \wedge L=1 \wedge Q=0 \rightarrow ab(A_1), & M=1 \wedge N=1 \wedge P=0 \rightarrow ab(A_2), \\
K=0 \wedge \qquad\quad Q=1 \rightarrow ab(A_1), & M=0 \wedge \qquad\quad P=1 \rightarrow ab(A_2), \\
\qquad\quad L=0 \wedge Q=1 \rightarrow ab(A_1), & \qquad\quad N=0 \wedge P=1 \rightarrow ab(A_2), \\[1em]
P=0 \wedge Q=0 \wedge S=1 \rightarrow ab(O_1), & \\
P=1 \wedge \qquad\quad S=0 \rightarrow ab(O_1), & \\
\qquad\quad Q=1 \wedge S=0 \rightarrow ab(O_1)\}. &
\end{array}
$$

Also, for all the queries that follow, we will assume the set $PI(SD)$ is available through compilation. Consider a scenario where the following values are observed on the wires (as shown in figure 1): $K=1, L=0, M=1, R=1$ and $S=0$ The observation is encoded in the form

$$K=1 \wedge L=0 \wedge M=1 \quad \rightarrow \quad R=1 \wedge S=0 \qquad \text{(OBS.1)}$$

as in *input* → *output*. The task is to find the assumption based *minimal explanation* E for OBS with repsect to SD i.e., $SD \cup E \models OBS$ and $SD \cup E$ is consistent. The set of all such E is:

$$
\begin{array}{ll}
ME(OBS.1,\mathcal{T}) = \{ & ab(X_1) \wedge \neg ab(X_2) \wedge \neg ab(A_1) \wedge \neg ab(A_2) \wedge \neg ab(O_1), \\
& ab(X_1) \wedge ab(A_1) \wedge ab(O_1) \wedge \neg ab(X_2), \\
& ab(X_1) \wedge ab(A_1) \wedge ab(O_2) \wedge \neg ab(X_2), \\
& ab(X_2) \wedge ab(A_2) \wedge \neg ab(X_1) \wedge \neg ab(A_1) \wedge \neg ab(O_1), \\
& ab(X_2) \wedge ab(O_1) \wedge \neg ab(X_1) \wedge \neg ab(A_2), \\
& ab(X_2) \wedge ab(A_1) \wedge ab(O_1) \wedge \neg ab(X_1)\}.
\end{array}
$$

Each member of ME, for instance the fourth explanation in which gates X_2 and A_2 are abnormal with the rest of the gates normal, will explain the observation OBS.1.

To illustrate the idea of *relativized conclusion*, consider another observation such that

$$K=1 \wedge L=0 \rightarrow S=0 \qquad \text{(OBS.2)}$$

There are three assumption based minimal explanations for OBS.2, namely

$$
\begin{array}{ll}
ME(OBS.2,\mathcal{T}) = \{ & ab(X_1) \wedge \neg ab(A_1) \wedge \neg ab(A_2) \wedge \neg ab(O_1), \\
& ab(X_1) \wedge ab(A_2) \wedge ab(O_1), \\
& ab(A_1) \wedge ab(O_1)\}.
\end{array}
$$

Notice that these explanations do not mention the gate X_2 and this is simply because the gate X_2 is not the focus, and not related to the observation.

Let us proceed to demonstrate inquiry using *conditional explanation.* Assuming the observation is

$$K = 1 \wedge M = 1 \wedge \neg ab(A_2) \wedge ab(O_1) \quad \rightarrow \quad S = 0, \tag{OBS.3}$$

where not abnormal of gate A_2 and abnormal of gate O_1 is knowledge that we have postulated. The inquiry here is to investigate the outcome of such postulation in conjunction with the observation. For variety, we shall query for the assumption based prime conditional explanation only ignoring the trivial ones. Recall that a conditional explanation is an explanation in the form $Ant \rightarrow Assump$, where Ant is non-assumption based or simply facts. The following is the set of prime conditional explanations (PCE) for the above observation (OBS.3) with postulation.

$$\begin{array}{lll} PCE(OBS.3, \mathcal{T}) = \{ & L = 1 \rightarrow ab(X_1), & L = 0 \rightarrow \neg ab(X_1), \\ & L = 0 \rightarrow ab(A_1), & L = 1 \rightarrow \neg ab(A_1), \\ & R = 1 \rightarrow ab(X_2), & R = 0 \rightarrow \neg ab(X_2), \\ & ab(X_1) \wedge ab(A_1), & \neg ab(X_1) \wedge \neg ab(A_1), \\ & N = 1, & \\ & Q = 1, & \\ & P = 1\}. & \end{array}$$

Consider the first explanation saying that if $L = 1$, then the abnormality of the gate X_1 will explain the OBS. Let us trace through the circuit in figure 1. It shows that when $L = 1$, with the observation $K = 1$ and the explanation $ab(X_1)$, the output $N = 1$. Since the gate A_2 is normal by our postulation and $M = 1$ by observation, the wire $P = 1$. Finally, the gate O_1 is abnormal by our postulation and hence the output $S = 0$, regardless of the status of Q.

We now focus on the issue of finding *extension* generating subsets of assumptions. Recall that an extension generating subset is a maximal subset of assumptions that is consistent with the theory. Therefore, the extension generating subsets of SD alone are the set of all maximal subsets of assumptions that are consistent with SD. Since SD is encoded with complete knowledge, that is it contains descriptions of both normal and abnormal state of components, there are 2^5 maximal consistent subsets of assumptions. These subsets range from all 5 gates being normal to all 5 gates being abnormal.

To make the investigation more interesting, lets consider finding extension generating subsets of SD augmented with an observation. Let the observation be

$$K = 1 \wedge L = 0 \wedge M = 1 \quad \rightarrow \quad R = 1 \wedge S = 0, \tag{OBS.4}$$

and let $\Sigma = SD \cup OBS.4$. Since $PI(SD)$ has been computed, the set $PI(\Sigma)$ is computed incrementally using the incremental algorithm described in [10], that is $PI(PI(SD) \cup OBS)$. The procedure to find generating subsets involves three successive steps: (1) find all the minimal *conflict* sets with respect to Σ; (2) compute the minimal *hitting* sets from the conflict sets; and (3) extract the *extension* generating subsets of assumptions from the hitting sets.

Firstly, by lemma 4.4, E a is minimal conflict set of the α-theory if $E \in MI(\Sigma)$ and $\overline{E} \subseteq \mathcal{A}$. Thus, using the method stated by corollary 4.1, we obtain the following set of minimal conflict sets (MCS) for Σ.

$$\begin{aligned}
MCS(\Sigma) = \{\ & \{ab(X_1), \neg ab(X_1)\}, \\
& \{ab(X_2), \neg ab(X_2)\}, \\
& \{ab(A_1), \neg ab(A_1)\}, \\
& \{ab(A_2), \neg ab(A_2)\}, \\
& \{ab(O_1), \neg ab(O_1)\}, \\[1em]
& \{\neg ab(X_1), \neg ab(X_2)\}, \\
& \{ab(X_1), ab(X_2)\}, \\
& \{ab(A_1), \neg ab(O_1)\}, \\
& \{ab(X_2), \neg ab(A_2), \neg ab(O_1)\}, \\
& \{ab(A_2), \neg ab(X_2), \neg ab(O_1)\}, \\
& \{ab(X_1), ab(A_2), \neg ab(O_1)\}, \\
& \{\neg ab(X_1), \neg ab(A_2), \neg ab(O_1)\}, \\
& \{ab(O_1), \neg ab(X_2), \neg ab(A_1), \neg ab(A_2)\}, \\
& \{ab(X_2), ab(A_2), ab(O_1), \neg ab(A_1)\}, \\
& \{ab(X_1), ab(O_1), \neg ab(A_1), \neg ab(A_2)\}, \\
& \{ab(A_2), ab(O_1), \neg ab(X_1), \neg ab(A_1)\}\}.
\end{aligned}$$

Notice that the first five minimal conflict sets are simply the trivial ones, that is the minimal contradictions from the assumption set $\mathcal{A}$. The sixth conflict set says that both gates X_1 and X_2 being normal is inconsistent with Σ. Similarly, the seventh conflict set says that both gates X_1 and X_2 being abnormal is also inconsistent with Σ. This information reveals that gate X_1 being normal/abnormal precludes gate X_2 being normal/abnormal and vice versa. Subsequently, using the transformation method described for computing *hitting* sets (i.e. lemma 4.5 and its optimization), we obtain the following set of all minimal hitting sets (MHS) for Σ.

$$\begin{aligned}
MHS(\Sigma) = \{\ & \{\neg ab(O_1), \neg ab(A_2), \neg ab(A_1), \neg ab(X_2), ab(X_1)\}, \\
& \{\neg ab(O_1), \neg ab(A_2), \neg ab(A_1), ab(X_2), \neg ab(X_1)\}, \\
& \{\neg ab(O_1), \neg ab(A_2), ab(A_1), ab(X_2), \neg ab(X_1)\}, \\
& \{\neg ab(O_1), ab(A_2), \neg ab(A_1), \neg ab(X_2), ab(X_1)\},
\end{aligned}$$

$$\begin{aligned}&\{\neg ab(O_1), ab(A_2), \neg ab(A_1), ab(X_2), \neg ab(X_1)\},\\&\{\neg ab(O_1), ab(A_2), ab(A_1), \neg ab(X_2), ab(X_1)\},\\&\{ab(O_1), \neg ab(A_2), ab(A_1), \neg ab(X_2), ab(X_1)\},\\&\{ab(O_1), ab(A_2), ab(A_1), ab(X_2), \neg ab(X_1)\}\}.\end{aligned}$$

Finally, by theorem 4.2 an extension generating subset is simply the set difference of $\mathcal{A}$ from a minimal hitting set. Thus, the set of all extension generating subsets (EXT) for Σ are as follows:

$$\begin{aligned}EXT(\Sigma) = \{\ &\{\neg ab(X_1), ab(X_2), ab(A_1), ab(A_2), ab(O_1)\},\\&\{ab(X_1), \neg ab(X_2), ab(A_1), ab(A_2), ab(O_1)\},\\&\{ab(X_1), \neg ab(X_2), \neg ab(A_1), ab(A_2), ab(O_1)\},\\&\{\neg ab(X_1), ab(X_2), ab(A_1), \neg ab(A_2), ab(O_1)\},\\&\{ab(X_1), \neg ab(X_2), ab(A_1), \neg ab(A_2), ab(O_1)\},\\&\{\neg ab(X_1), ab(X_2), \neg ab(A_1), \neg ab(A_2), ab(O_1)\},\\&\{\neg ab(X_1), ab(X_2), \neg ab(A_1), ab(A_2), \neg ab(O_1)\},\\&\{ab(X_1), \neg ab(X_2), \neg ab(A_1), \neg ab(A_2), \neg ab(O_1)\}\ \}.\end{aligned}$$

Continuing with the example, we shall discuss the decision problems of *explainability*, *agreement* and *irrefutability*. An explainable observation is simply an observation that has an explanation or contrariwise, an *inexplicable OBS* is one that has no explanation. Consider the following observation

$$K = 1 \wedge L = 0 \wedge M = 1 \wedge ab(X_1) \quad \rightarrow \quad R = 1, \tag{OBS.5}$$

its minimal explanations are $\neg ab(X_1)$ and $\neg ab(X_2)$. Thus OBS.5 is explainable with respect to the SD and the assumption set $\mathcal{A}$. Consider the scenario that we are only interested in the positive assumptions restated as follows:

$$\mathcal{A}' = \{ab(X_1),\ ab(X_2),\ ab(A_1),\ ab(A_2),\ ab(O_1)\}.$$

Under this new set assumptions, there is no explanation for OBS.5 with respect to SD and $\mathcal{A}'$. An inexplicable observation may serve as information to the system designer that there are insufficient assumptions, or at the other extreme, as a strategy to focus attention on certain assumptions. Also note that OBS.5 is not *agreeable* with respect to SD and $\mathcal{A}$, that is the observation OBS.5 is explainable but the negation $\neg OBS.5$

$$K = 1 \wedge L = 0 \wedge M = 1 \wedge ab(X_1) \wedge R = 0 \tag{OBS.6}$$

is not. Intuitively the *agreement* property says that OBS.5 is consistent in some extension but there exist other extensions that the observation is not consistent with. One useful result would be to find these extensions that the observation is inconsistent with. We shall investigate a method using direct consequence later.

In verifying an *irrefutable* observation, we shall utilize the previous set $PI(\Sigma)$ from OBS.4 and the assumption set $\mathcal{A}$ to demonstarate the idea. Consider the following observation

$$ab(X_1) \rightarrow ab(X_2) \quad \wedge \quad ab(X_2) \rightarrow ab(X_1) \qquad \text{(OBS.7)}$$

suggesting that the abnormality of either X_1 or X_2 implies the abnormality of the other. The minimal explanations for OBS.7 are

$$\begin{aligned} ME(OBS.7, (\Sigma, \mathcal{A})) = \{ \; & ab(X_1) \wedge ab(X_2), \\ & ab(X_1) \wedge \neg ab(O_1) \wedge ab(A_2), \\ & ab(X_1) \wedge ab(O_1) \wedge \neg ab(A_2) \wedge \neg ab(A_1), \\ & \neg ab(X_2) \wedge \neg ab(X_1), \\ & \neg ab(X_2) \wedge \neg ab(O_1) \wedge ab(A_2), \\ & \neg ab(X_2) \wedge ab(O_1) \wedge \neg ab(A_2) \wedge \neg ab(A_1), \\ & \neg ab(A_2) \wedge \neg ab(O_1) \wedge ab(X_2), \\ & \neg ab(A_2) \wedge \neg ab(O_1) \wedge \neg ab(X_1), \\ & ab(O_1) \wedge ab(A_2) \wedge \neg ab(A_1) \wedge ab(X_2), \\ & ab(O_1) \wedge ab(A_2) \wedge \neg ab(A_1) \wedge \neg ab(X_1) \; \}. \end{aligned}$$

According to theorem 3.2, if the conjunction of the negation of these minimal explanations is not explainable with respect to $(\Sigma, \mathcal{A})$, then OBS.7 is *irrefutable*. In fact, irrefutablility of an observation implies that it is consistent with all extensions, or more intuitively it is explainable in every extension. The reader can verify the irrefutability of OBS.7 by comparing the above explanations to the extensions generated earlier for OBS.4.

One feature of *direct consequence* is that it allows the system to find the *prime* conflict sets of a given observation with respect to SD, modulo assumptions. Simple propositional reasoning will show that this is true. As in the definition of direct consequence (definition 3.6), $SD \cup OBS \models C$ and by definition of conflict set (definition 4.2), the set $\overline{C}$ is a conflict set for $SD \cup OBS$. Since a direct consequence C has the property that $SD \not\models C$, therefore this set $\overline{C}$ is a non-trivial conflict set, that is the set $\overline{C}$ is not a contradiction by itself. By virtue of the minimality of a direct consequence, this set $\overline{C}$ is therefore a prime conflict set for $SD \cup OBS$.

Moreover using these prime conflict sets and constructing trivial ones on-the-fly, we can compute the extension generating subsets with respect to $SD{\cup}OBS$, without actually computing the prime implicates of it as it was done in OBS.4. For example, the minimal direct consequences of OBS.6 are $ab(X_1)$ and $ab(X_2)$ and hence the minimal conflict sets with respect to $SD\cup$ OBS.6 are

$$\begin{aligned} MCS(SD\cup \text{ OBS.6}) = \{ \ & \{\neg ab(X_2)\}, \\ & \{\neg ab(X_1)\}, \\ & \{\neg ab(O_1), ab(O_1)\}, \\ & \{\neg ab(A_2), ab(A_2)\}, \\ & \{\neg ab(A_1), ab(A_1)\}\}. \end{aligned}$$

And using the same method of computing extensions generatting subsets, the set of generating subsets for $SD\cup$ OBS.6 is

$$\begin{aligned} EXT(SD\cup\text{OBS.6}) = \{ \quad & \{ab(X_1), ab(X_2), ab(A_1), ab(A_2), ab(O_1)\}, \\ & \{ab(X_1), ab(X_2), ab(A_1), ab(A_2), \neg ab(O_1)\}, \\ & \{ab(X_1), ab(X_2), ab(A_1), \neg ab(A_2), ab(O_1)\}, \\ & \{ab(X_1), ab(X_2), ab(A_1), \neg ab(A_2), \neg ab(O_1)\}, \\ & \{ab(X_1), ab(X_2), \neg ab(A_1), ab(A_2), ab(O_1)\}, \\ & \{ab(X_1), ab(X_2), \neg ab(A_1), ab(A_2), \neg ab(O_1)\}, \\ & \{ab(X_1), ab(X_2), \neg ab(A_1), \neg ab(A_2), ab(O_1)\}, \\ & \{ab(X_1), ab(X_2), \neg ab(A_1), \neg ab(A_2), \neg ab(O_1)\} \ \}. \end{aligned}$$

Note that the extensions of OBS.6, which is the negation of OBS.5, will be the extensions that are inconsistent with OBS.5 with respect to the same SD and $\mathcal{A}$.

Another interesting usage of direct consequence is to vary the definition by interchanging the role of assumptions between the observation and the direct consequence. That is, the observation is comprised solely of assumption literals, and the converse for direct consequence. Effectively we have a definition of *prediction*, that is under the observed assumption, we compute the most direct outcome (consequence), with repsect to SD and $\mathcal{A}$. This is merely a hint of the vast range of applications for assumption based reasoning.

6 Conclusions

In this paper, we have argued that in the realm of *truth maintenance systems*, the subject matter should be studied using a formal methodology. As a consequence, the formal aspects of the specification of truth maintenance suggests that there is this notion of assumption based reasoning in general. We have explored an assumption based reasoning theory with the notion of direct consequence, explanation, conflict set, extension, agreement and irrefutability. These are just some possibilities for the many more that have yet to be explored. We have also provided a computational system (*ACMS*) that performs the computations of the above functions. As illustrated in the Boolean circuit example, advances in the study of assumption based reasoning could make the mechanization of the logic of question and answer a reality.

Acknowledgement: We are very grateful to Jane Mulligan, Paul Gilmore and Alan Mackworth for their comments and criticism.

References

[1] T.C. Bartee, I.L. Lebow, and I.S. Reed. *Theory and Design of Digital Machines.* McGraw-Hill Book, 1962.

[2] Jan Berg. *Bolzano's Logic.* Almqvist and Wiksell, Stockholm, 1962.

[3] Wolfgang Bibel. *Automated Theorem Proving.* Vieweg, Braunschweig, second edition, 1987.

[4] Phillip T. Cox and Thomas Pietrzykowski. Causes for events: Their computations and applications. In *8th Conference on Automated Deduction*, pages 608–621, Oxford, England, 1986.

[5] Martin Davis. The mathematics of non-monotonic reasoning. *Artificial Intelligence*, 13, 1980.

[6] Johan de Kleer. An assumption-based tms. *Artificial Intelligence*, 28, 1986.

[7] Johan de Kleer. Extending the atms. *Artificial Intelligence*, 28, 1986.

[8] Jon Doyle. A truth maintenance system. *Artificial Intelligence*, 12, 1979.

[9] J. J. Finger and M. R. Genesereth. Residue: A deductive approach to design synthesis. Memo HPP 1, Department of Computer Science, Stanford University, 1985. Stanford Heuristic Programming Project.

[10] Alex Kean and George Tsiknis. An incremental method for generating prime implicants/implicates. *Journal of Symbolic Computation*, 1990. to appear.

[11] J. P. Martin and S. C. Shapiro. A model for belief revision. *Artificial Intelligence*, 35, 1988.

[12] David Poole, Randy Goebel, and Roman Aleliunas. Theorist: A logical reasoning system for defaults and diagnosis. CS Research Report 6, Department of Computer Science, University of Waterloo, 1986. Logic Programming and Artificial Intelligence Group.

[13] Allan Ramsey. *Formal Methods in Artificial Intelligence.* Cambridge University Press, 1988.

[14] Raymond Reiter. A theory of diagnosis from first principle. *Artificial Intelligence*, 32, 1987.

[15] Raymond Reiter and Johan de Kleer. Foundations of assumption-based truth maintenance systems: Preliminary report. In *Proceeding of AAAI-87*, pages 183–188, Seatle, Washington, 1987.

[16] Joseph R. Shoenfield. *Mathematical Logic.* Addison-Wesley, 1967.

[17] George Tsiknis and Alex Kean. Clause management systems (cms). Technical Report 21, Department of Computer Science, University of British Columbia, 1988.

[18] Johan van Benthem. The variety of consequence, according to bolzano. *STUDIA LOGICA*, XLIV(4), 1985.