

# Assumption-Based Reasoning and Model-Based Diagnostics \*

R. Haenni, P.A. Monney, J. Kohlas  
Institute of Informatics  
University of Fribourg  
Regina Mundi  
CH-1700 Fribourg  
Switzerland

Phone: (+41 37) 29 83 20

Fax: (+41 37) 29 97 27

E-Mail: Juerg.Kohlas@unifr.ch

March 30, 1995

## Abstract

This paper shows how assumption-based reasoning can be used to evaluate the hypothesis that a particular set of components of a system is not working properly. Like in model-based diagnostics (Reiter, 1987), the behaviour of the system is described by a set of propositional logic formulas. These formulas contain a special type of propositions called assumptions which indicate whether or not a particular component is working properly. Then, under some assumptions, the knowledge base will permit to conclude the validity of some hypotheses regarding the operating mode of certain components. This paper

---

\*Research supported by grants No. 21-30186.90 and 21-32660.91 of the Swiss National Foundation for Research, Esprit Basic Research Activity Project DRUMS II (Defeasible Reasoning and Uncertainty Management).

presents several new results about the representation of these sets of assumptions. Then, in a second step, probabilities on the assumptions are introduced in order to obtain a quantitative evaluation of the credibility of the hypotheses.

Using a simple example, Section 1 quickly introduces the main concepts and ideas and Section 2 develops the general theory and presents the new results. Section 3 shows how probabilities can be used to add a quantitative perspective to the evaluation of hypotheses. Finally, Section 4 discusses two examples: a binary adder and a sequence of inverters.

## 1 An Introductory Example

The easiest way to get familiar with the idea of assumption-based reasoning is to discuss a very simple example. Consider a sequence of two binary inverters as pictured in Fig. 1.

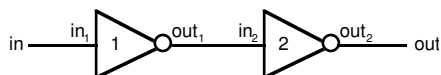


Figure 1: The logical circuit of two inverters.

When working properly, a binary inverter returns the opposite of its input. In the system pictured in Fig. 1 we can only observe the input variable  $in$  and the output variable  $out$  and suppose that we observe  $out = 1$  when  $in = 0$ . Then it can be concluded that it is impossible for both inverters to be working properly. Suppose that it is known that the a priori probability that inverter  $i$  is working properly equals  $p_i, i = 1, 2$ . Let  $h$  denote the hypothesis that the inverter  $i$  is broken. The problem is then to determine to what extent  $h$  is supported by the observations we have made and our knowledge about the general functioning of the system.

To answer this question, the first step is to describe the system by logical formulas: the atoms are the 6 propositions  $in, in_1, \dots$  that appear in Fig. 1 and the two special propositions  $ok_1$  and  $ok_2$  representing the assumptions that the inverters 1 and 2 are working properly. If we assume that a broken inverter always returns the opposite of what it should return (i.e. the output

is the same as the input), then the formulas

$$\begin{aligned} ok_1 &\rightarrow (out_1 \leftrightarrow \neg in_1), \quad \neg ok_1 \rightarrow (out_1 \leftrightarrow in_1), \\ ok_2 &\rightarrow (out_2 \leftrightarrow \neg in_2), \quad \neg ok_2 \rightarrow (out_2 \leftrightarrow in_2) \end{aligned} \quad (1)$$

certainly belong to the knowledge base. It is also necessary to represent the connection between the inverters, as well as the fact that  $in = in_1$  and  $out = out_2$ :

$$in \leftrightarrow in_1, \quad out_1 \leftrightarrow in_2, \quad out_2 \leftrightarrow out. \quad (2)$$

Finally, the fact that we observe  $in = 0$  and  $out = 1$  must be added to the knowledge base:

$$\neg in, \quad out. \quad (3)$$

Then, under some assumptions yet to be determined, the knowledge base will permit to logically infer that the inverter 1 is broken, which establishes the validity of the hypothesis that the inverter 1 is broken. If such a collection of assumptions is regrouped in a set  $S$ , then the conjunction of all assumptions in  $S$  is called a *quasi-support term* of the hypothesis  $h$ . The disjunction of all quasi-support terms of  $h$  is called the *quasi-support* of  $h$ . In this example, there are two quasi-support terms of the hypothesis  $\neg ok_1$  (i.e. the inverter 1 is broken), namely  $\neg ok_1$  and  $ok_2$ . Therefore the quasi-support of  $\neg ok_1$  is  $\neg ok_1 \vee ok_2$ . The quasi-support terms of the contradictory hypothesis  $\perp$  represent impossible configurations of the system and as this paper will show, they play a very important role in the representation of the quasi-support of any hypothesis. In this example there are two of them:  $ok_1 \wedge ok_2$  and  $\neg ok_1 \wedge \neg ok_2$ . Since a quasi-support term of the contradiction is always also a quasi-support of any hypothesis  $h$ , it is not really supporting the hypothesis  $h$  since it does so only because it is in contradiction with the knowledge base. Therefore we define the *support* of  $h$  as the conjunction of the quasi-support of  $h$  and the negation of the quasi-support of the contradiction. In this example, the support of  $\neg ok_1$  is  $\neg ok_1 \wedge ok_2$ . The support and quasi-support of a hypothesis  $h$  give a qualitative perspective to the question as whether  $h$  is true or not. Considering the a priori probabilities on the assumptions will introduce a quantitative aspect in the evaluation of the hypothesis. This quantitative aspect is represented by the so-called *degree of support* of the hypothesis  $h$ . It is denoted by  $sp(h)$ . The next section will give a natural

and precise definition of the degree of support. In this example, if we define  $q_i = 1 - p_i, i = 1, 2$ , then it can be proved that the degree of support that the inverter 1 is broken equals

$$sp(\neg ok_1) = \frac{p_2 q_1}{p_2 q_1 + p_1 q_2} \quad (4)$$

and the degree of support that the inverter 2 is broken equals

$$sp(\neg ok_2) = \frac{p_1 q_2}{p_2 q_1 + p_1 q_2}. \quad (5)$$

## 2 Fundamental Notions and Results

Let's start by reviewing some basic elements of propositional logic that will be used in the sequel. If  $P = \{p_1, \dots, p_n\}$  is a set of propositional symbols, then  $\mathcal{L}_P$  denotes the set of all logical formulas using symbols in  $P$ : the language over  $P$ . Also, let  $B_P = \{0, 1\}^{|P|}$  denote the Boolean cube of all possible interpretations of formulas in  $\mathcal{L}_P$ : an interpretation of a formula is an assignment of truth values (i.e. 0 and 1) to the propositional symbols occurring in the formula. Given a formula  $f \in \mathcal{L}_P$  and an interpretation  $x \in B_P$ , we can determine the evaluation of  $f$  under the interpretation  $x$ : this defines a function  $ev : \mathcal{L}_P \rightarrow \{0, 1\}$ . This in turn permits to define the set of so-called models of a formula, namely

$$N(f) = \{x \in B_P : ev(f, x) = 1\}. \quad (6)$$

Two formulas  $f$  and  $g$  in  $\mathcal{L}_P$  are equal if  $N(f) = N(g)$ . A literal of the set of propositions  $P = \{p_1, \dots, p_n\}$  is an element of  $P$  or its negation. Now let  $M_P$  denote the set of all conjunctions of literals of  $P$  having exactly  $n$  different symbols, i.e.

$$M_P = \{\wedge_{i=1}^n l_i : l_i = p_i \text{ or } l_i = \neg p_i\}. \quad (7)$$

Of course, there is a one-to-one correspondance between the elements of  $B_P$  and those of  $M_P$ . For  $x \in B_P$ , let  $\sigma(x)$  denote the unique element of  $M_P$  such that  $ev(\sigma(x), x) = 1$ . Finally, a formula  $f$  is a logical consequence of another formula  $g$ , written  $g \models f$ , if and only if  $N(f) \subseteq N(g)$ . So  $f = g$  if and only if  $f \models g$  and  $g \models f$ .

Equipped with these notions from propositional logic, we are now in a position to develop the general theory corresponding to the ideas intuitively presented in Section 1. Let  $A = \{a_1, \dots, a_s\}$  denote a set of propositional symbols called assumptions and let  $P = \{p_1, \dots, p_n\}$  be another, disjoint set of propositional symbols. Let  $\Sigma = \{\xi_1, \dots, \xi_m\}$  denote a set of formulas in  $\mathcal{L}_{A+P}$ , which we assume without loss of generality to be clauses (any formula can be transformed into an equivalent conjunction of clauses). Like in model-based diagnostics (Reiter, 1987), the set  $\Sigma$  represents the knowledge base. We are particularly interested in certain hypotheses expressed as formulas  $h$  in  $\mathcal{L}_{A+P}$ . If some assumptions are assumed to be true, then  $h$  can possibly be deduced from  $\xi = \xi_1 \wedge \dots \wedge \xi_m$ . So we may ask, under what assumptions can  $h$  be inferred from  $\xi$ ? Defining  $\mathcal{C}_A$  as the set of all conjunctions of 0, 1, 2, etc conjunctions of literals in  $A$ , it is then very natural to look for all  $a \in \mathcal{C}_A$  such that  $a \wedge \xi \models h$ . Such a conjunction  $a$  is called a *quasi-support term* of  $h$  and the formula

$$Sp'(h) = \vee \{a \in \mathcal{C}_A : a \wedge \xi \models h\} \quad (8)$$

is called the *quasi-support* of  $h$ . A quasi-support term  $a$  of  $\perp$  is called a contradictory quasi-support term because  $a \wedge \xi = \perp$ , which means that  $a$  is in contradiction with the knowledge base. If  $a$  is a contradictory quasi-support term, then  $a \wedge \xi \models \perp \models h$  for every formula  $h \in \mathcal{L}_P$ . This means that a contradictory quasi-support term is a quasi-support term of all formulas. However, such a quasi-support term is not a true argument in favour of  $h$  since it permits to infer  $h$  only because it is in contradiction with the knowledge base. For this reason, the formula

$$Sp(h) = Sp'(h) \wedge \neg Sp'(\perp) \quad (9)$$

is called the *support* of  $h$ .

It is clear that a conjunction  $a \in \mathcal{C}_A$  such that  $a \wedge \xi \models \neg h$  speaks against the hypothesis  $h$ , i.e.  $a$  sheds doubt on the hypothesis  $h$ . So we define the *doubt* of  $h$  by

$$D(h) = \vee \{a \in \mathcal{C}_A : a \wedge \xi \models \neg h\} \quad (10)$$

and the *plausibility* of  $h$  by

$$Pl(h) = \neg D(h) \quad (11)$$

for all  $h \in \mathcal{L}_{A+P}$ . Of course these definitions imply that  $D(h) = Sp'(\neg h)$  and hence

$$Pl(h) = \neg Sp'(\neg h). \quad (12)$$

for all  $h \in \mathcal{L}_{A+P}$ .

It can be proved (Kohlas, 1994) that  $Sp' : \mathcal{L}_{A+P} \rightarrow \mathcal{L}_A$  is a so-called *allocation of support*, which means that

$$\begin{aligned} Sp'(h_1 \wedge h_2) &= Sp'(h_1) \wedge Sp'(h_2) \\ Sp'(\top) &= \top. \end{aligned} \quad (13)$$

It is easy to see that this implies the following properties

$$\begin{aligned} \text{If } h_1 \models h_2 \text{ then } Sp'(h_1) &\models Sp'(h_2), \\ Sp'(h_1) \vee Sp'(h_2) &\models Sp'(h_1 \vee h_2). \end{aligned} \quad (14)$$

For more information about allocations of supports, e.g. the combination of allocations of supports, see (Kohlas, 1994).

The quasi-support of a formula  $h$  in  $\mathcal{L}_{A+P}$  ( $h$  can be the contradiction) can be obtained by advanced consequence finding algorithms developed by (Siegel, 1987) and (Inoue, 1992) (see also (Reiter & de Kleer, 1987)). For a presentation of these methods in the context of assumption-based reasoning, see (Kohlas & Monney, 1994). Besides this traditional approach, there is another way to obtain the quasi-support of a formula. This new technique utilizes the valuation-based systems developed by (Shenoy, 1994) and is explained in (Kohlas, 1993a) (for a short version, see (Kohlas, 1993b)). These two techniques have been successfully implemented at the Institute of Informatics of the University of Fribourg.

The following theorem is the main result of the paper. In particular, the first equation shows that the quasi-support of any formula  $h$  in  $\mathcal{L}_A$  can be readily obtained from the quasi-support of the contradiction. This means that when the hypothesis is expressed as a formula over the assumptions, the only thing to be computed is the quasi-support of the contradiction. Of course, the two general techniques mentioned above can be applied to compute the quasi-support of the contradiction only, and this is much more efficient than applying these techniques to compute the quasi-support of the formula  $h$  itself.

**Theorem 1** *If  $h$  is a formula in  $\mathcal{L}_A$ , then*

$$Sp'(h) = h \vee Sp'(\perp), \quad (15)$$

$$Sp(h) = h \wedge \neg Sp'(\perp), \quad (16)$$

$$Pl(h) = Sp(h). \quad (17)$$

*Proof*

In order to prove this theorem, we need the following lemma.

**Lemma 1** *Let  $a \in \mathcal{C}_A$  be an arbitrary conjunction of literals in  $A$ . Then*

$$Sp'(a) = a \vee Sp'(\perp). \quad (18)$$

*Proof*

We must prove that

$$N(Sp'(a)) = N(a) \cup N(Sp'(\perp)). \quad (19)$$

First note that

$$N(a) = \{x \in B_A : ev(a, x) = 1\} = \{x \in B_A : \sigma(x) \models a\} \quad (20)$$

because  $ev(a, x) = 1$  if and only if  $\sigma(x)$  contains  $a$  as a subconjunction, which is true if and only if  $\sigma(x) \models a$ . Since it can easily be proved that

$$Sp'(h) = \vee \{t \in M_A : t \wedge \xi \models h\} \quad (21)$$

for all  $h \in \mathcal{L}_{A+P}$ , it follows that

$$\begin{aligned} N(Sp'(h)) &= \{x \in B_A : ev(Sp'(h), x) = 1\} \\ &= \{x \in B_A : ev(\vee \{t \in M_A : t \wedge \xi \models h\}, x) = 1\} \\ &= \{x \in B_A : \vee \{ev(t, x) : t \wedge \xi \models h\} = 1\} \\ &= \{x \in B_A : \sigma(x) \wedge \xi \models h\} \end{aligned} \quad (22)$$

and hence in particular

$$N(Sp'(a)) = \{x \in B_A : \sigma(x) \wedge \xi \models a\}, \quad (23)$$

$$N(Sp'(\perp)) = \{x \in B_A : \sigma(x) \wedge \xi = \perp\}. \quad (24)$$

Now let's prove that

$$N(Sp'(a)) \subseteq N(a) \cup N(Sp'(\perp)). \quad (25)$$

Let  $x \in B_A$  such that  $\sigma(x) \wedge \xi \models a$ . We consider two cases:

1. Suppose that  $\sigma(x) \models a$ . Then  $ev(a, x) = 1$  and hence  $x \in N(a)$ .
2. Suppose that  $\sigma(x) \not\models a$ . We are going to prove that  $\sigma(x) \wedge \xi = \perp$  and hence  $x \in N(Sp'(\perp))$ . Suppose that there is a vector  $v \in B_{A+P}$  such that  $ev(\sigma(x) \wedge \xi, v) = 1$ . Then  $ev(\sigma(x), v) = 1$  and hence there is a vector  $r \in B_P$  such that  $v = (x, r)$ . But by hypothesis we have also  $ev(a, v) = 1$  and hence  $ev(a, x) = 1$  because  $v = (x, r)$  and therefore  $\sigma(x) \models a$ , which is a contradiction.

Now let's prove that

$$N(a) \cup N(Sp'(\perp)) \subseteq N(Sp'(a)). \quad (26)$$

If  $x \in N(a)$  then  $\sigma(x) \models a$  and hence  $\sigma(x) \wedge \xi \models a$  and hence  $x \in N(Sp'(a))$ . If  $x \in N(Sp'(\perp))$  then  $\sigma(x) \wedge \xi = \perp$  and hence  $\sigma(x) \wedge \xi \models a$  and hence  $x \in N(Sp'(a))$ .

◇

Before we prove equation 15, we need another intermediate result. Let  $a = \bigwedge_{i=1}^m l_i$  and  $a' = \bigwedge_{j=1}^n l'_j$  be two elements in  $\mathcal{C}_A$ . Then by the distributivity law and the fact that  $Sp'$  is an allocation of support, we have

$$\begin{aligned} Sp'((\bigwedge_{i=1}^m l_i) \vee (\bigwedge_{j=1}^n l'_j)) &= Sp'(\bigwedge_{i,j} (l_i \vee l'_j)) \\ &= \bigwedge_{i,j} (Sp'(l_i) \vee Sp'(l'_j)) \\ &= (\bigwedge_{i=1}^m Sp'(l_i)) \vee (\bigwedge_{j=1}^n Sp'(l'_j)) \\ &= Sp'(\bigwedge_{i=1}^m l_i) \vee Sp'(\bigwedge_{j=1}^n l'_j) \\ &= Sp'(a) \vee Sp'(a') \end{aligned} \quad (27)$$

and therefore

$$Sp'(a \vee a') = Sp'(a) \vee Sp'(a'). \quad (28)$$

Now let's prove equation 15. Since any formula in  $\mathcal{L}_A$  can be transformed into an equivalent disjunctive normal form, there exist conjunctions of literals  $f_1, \dots, f_n$  in  $\mathcal{L}_A$  such that  $f = f_1 \vee \dots \vee f_n$ . Then by equation 28 and lemma 1 we have

$$\begin{aligned} Sp'(f) &= Sp'(f_1 \vee \dots \vee f_n) \\ &= Sp'(f_1) \vee \dots \vee Sp'(f_n) \\ &= f_1 \vee Sp'(\perp) \vee \dots \vee f_n \vee Sp'(\perp) \\ &= f \vee Sp'(\perp). \end{aligned} \quad (29)$$

The proof of 16 is a consequence of 15:

$$\begin{aligned}
Sp(h) &= Sp'(h) \wedge \neg Sp'(\perp) \\
&= (h \vee Sp'(\perp)) \wedge \neg Sp'(\perp) \\
&= (h \wedge \neg Sp'(\perp)) \vee (Sp'(\perp) \wedge \neg Sp'(\perp)) \\
&= h \wedge \neg Sp'(\perp).
\end{aligned} \tag{30}$$

The proof of equation 17 is a consequence of equations 12 and 15:

$$\begin{aligned}
Pl(h) &= \neg Sp'(\neg h) \\
&= \neg(\neg h \vee Sp'(\perp)) \\
&= h \wedge \neg Sp'(\perp) = Sp(h).
\end{aligned} \tag{31}$$

◇

### 3 Computing Degrees of Support

In a second stage of the analysis, probabilities over the assumptions are introduced. So let  $P(a_i)$  denote the known probability that the assumption  $a_i$  holds. Assuming stochastic independence between assumptions, this generates a product probability  $P$  on  $M_A$ :

$$P(\bigwedge_{i=1}^s l_i) = \prod_{l_i=a_i} P(a_i) \cdot \prod_{l_i=\neg a_i} (1 - P(a_i)). \tag{32}$$

Since the conjunctions in  $M_A$  and the Boolean vectors in  $B_A$  are in a one-to-one correspondance, the probability  $P$  can also be considered as a probability of  $B_A$ . This in turn permits to define the probability  $P(f)$  of any formula  $f \in \mathcal{L}_A$  by

$$P(f) = P(N(f)). \tag{33}$$

So we define the *degree of quasi-support* of a formula  $h \in \mathcal{L}_{A+P}$  by

$$sp'(h) = P(N(Sp'(h))) \tag{34}$$

and the *degree of support* of  $h \in \mathcal{L}_{A+P}$  by the conditional probability of  $Sp(h)$  given  $\neg Sp'(\perp)$ :

$$sp(h) = \frac{P(N(Sp(h)))}{P(N(\neg Sp'(\perp)))}. \tag{35}$$

In a similar way, we also define the degree of plausibility of  $h \in \mathcal{L}_{A+P}$  by

$$pl(h) = \frac{P(N(Pl(h)))}{P(N(\neg Sp'(\perp)))}. \quad (36)$$

Since by equation 17  $Sp(h) = Pl(h)$  it follows immediately that

$$sp(h) = pl(h) \quad (37)$$

for all  $h \in \mathcal{L}_A$ . According to equation 16, equation 35 shows that the degree of support of a hypothesis  $h \in \mathcal{L}_A$  is simply the conditional probability of  $h$  given non-contradiction:

$$sp(h) = \frac{P(N(h \wedge \neg Sp'(\perp)))}{P(N(\neg Sp'(\perp)))}. \quad (38)$$

Since  $N(h \wedge \neg Sp'(\perp)) = N(h) - N(Sp'(\perp))$ , equation 38 implies that

$$sp(h) = \frac{P(N(h)) - P(N(Sp'(\perp)))}{1 - P(N(Sp'(\perp)))} \quad (39)$$

for all  $h \in \mathcal{L}_A$ . This shows that we need a general and efficient procedure to compute the probability of an arbitrary formula  $f \in \mathcal{L}_A$ .

To do this, the first step is to transform  $f$  into an equivalent disjunctive normal form (DNF), i.e. find  $f'_1, \dots, f'_m \in \mathcal{C}_A$  such that  $f = f'_1 \vee \dots \vee f'_m$ . Such a transformation is always possible and the classical algorithms can be used. Then, using the algorithm of Abraham (1979) or the algorithm of Heidtmann (1989), we can find conjunctions  $f_1, \dots, f_n$  in  $\mathcal{L}_A$  such that

$$f = f_1 + \dots + f_n \quad (40)$$

where the  $f_i$  are disjoint, i.e.  $f_i \wedge f_j = \perp$  whenever  $f_i \neq f_j$ . Then the probability of  $f$  is easily obtained by

$$P(N(f)) = \sum_{i=1}^n P(N(f_i)), \quad (41)$$

and  $P(N(f_i))$  is easy to compute because  $f_i$  is a conjunction of literals (e.g. if  $f_i = a_2 \wedge \neg a_3$  then  $P(N(f_i)) = P(a_2)(1 - P(a_3))$ ).

The next theorem shows how the degree of support of a formula  $h \in \mathcal{L}_{A+P}$  can be expressed in terms of degrees of quasi-supports:

**Theorem 2** If  $h \in \mathcal{L}_{A+P}$ , then

$$sp(h) = \frac{sp'(h) - sp'(\perp)}{1 - sp'(\perp)}. \quad (42)$$

*Proof*

$$\begin{aligned} sp(h) &= \frac{P(N(Sp(h)))}{1 - P(N(Sp'(h)))} \\ &= \frac{P(N(Sp'(h) \wedge \neg Sp'(\perp)))}{1 - sp'(h)} \\ &= \frac{P(N(Sp'(h)) - N(Sp'(\perp)))}{1 - sp'(\perp)} \\ &= \frac{P(N(Sp'(h)) - P(N(Sp'(\perp)))}{1 - sp'(\perp)} \\ &= \frac{sp'(h) - sp'(\perp)}{1 - sp'(\perp)}. \end{aligned} \quad (43)$$

◇

Once  $Sp'(h)$  and  $Sp'(\perp)$  have been found, the procedure described above can be used to compute  $sp'(h)$  and  $sp'(\perp)$  and then finally obtain  $sp(h)$  by formula 42. However, when  $h$  is in  $\mathcal{L}_A$ , it is possible to be more efficient. First find a DNF representation for  $Sp'(\perp)$  and  $h$ , i.e.

$$\begin{aligned} Sp'(\perp) &= f'_1 \vee \dots \vee f'_{m'} \\ h &= g'_1 \vee \dots \vee g'_{n'}. \end{aligned} \quad (44)$$

Then by equation 15 we have

$$\begin{aligned} Sp'(h) &= h \vee Sp'(\perp) \\ &= f'_1 \vee \dots \vee f'_{m'} \vee g'_1 \vee \dots \vee g'_{n'}. \end{aligned} \quad (45)$$

In the next step, apply the algorithm of Abraham or the algorithm of Heitmann on  $f'_1, \dots, f'_{m'}$  to obtain conjunctions  $f_1, \dots, f_m$  such that

$$Sp'(\perp) = f_1 + \dots + f_m \quad (46)$$

and then, because of equation 45, continue the algorithm with  $g'_1, \dots, g'_{n'}$  to obtain additional conjunctions  $g_1, \dots, g_n$  such that finally

$$Sp'(h) = f_1 + \dots + f_m + g_1 + \dots + g_n. \quad (47)$$

This implies that

$$\begin{aligned}
sp'(h) - sp'(\perp) &= P(N(Sp'(h))) - P(N(Sp'(\perp))) \\
&= \sum_{i=1}^m P(N(f_i)) + \sum_{i=1}^n P(N(g_i)) - \sum_{i=1}^m P(N(f_i)) \\
&= \sum_{i=1}^n P(N(g_i))
\end{aligned} \tag{48}$$

and hence

$$sp(h) = \frac{\sum_{i=1}^n P(N(g_i))}{1 - \sum_{i=1}^m P(N(f_i))} \tag{49}$$

according to theorem 2.

## 4 Examples

The results of Section 3 are especially important in diagnostic applications where the assumptions describe whether the components are working or not. The problem is to find possible explanations for an incorrect system behaviour and to judge hypotheses composed of assumptions. According to theorem 1, the main task is then the computation of the quasi-support of the contradiction. This section discusses two examples of simple digital circuits.

### 4.1 Cascaded Inverters

Consider the example of cascaded inverters of Section 1. Figure 2 shows a similar system with 4 binary inverters. Again, suppose that when the input is 0 we observe that the output is 1.

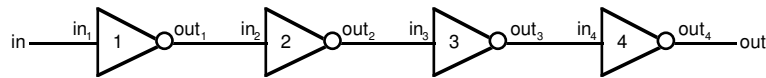


Figure 2: The circuit of 4 cascaded inverters.

A correctly working inverter gate produces a 0 when the input is 1 and it produces 1 when the input is 0. It is easy to see that in our series of 4 inverters (gates) there must be at least one faulty inverter. First, let's assume

that a faulty inverter precisely returns the opposite of what it should. This is a possible way to describe a faulty inverter, but this is not the only one. We could also assume that a faulty component always returns 0, or that for a broken gate every result (even the correct one) is possible. The case where a faulty inverter always produces the opposite of what it should can be transformed into the following set of logical formulas:

$$\begin{aligned}
ok_1 &\rightarrow (in_1 \leftrightarrow \neg out_1), & \neg ok_1 &\rightarrow (in_1 \leftrightarrow out_1), \\
ok_2 &\rightarrow (in_2 \leftrightarrow \neg out_2), & \neg ok_2 &\rightarrow (in_2 \leftrightarrow out_2), \\
ok_3 &\rightarrow (in_3 \leftrightarrow \neg out_3), & \neg ok_3 &\rightarrow (in_3 \leftrightarrow out_3), \\
ok_4 &\rightarrow (in_4 \leftrightarrow \neg out_4), & \neg ok_4 &\rightarrow (in_4 \leftrightarrow out_4).
\end{aligned} \tag{50}$$

The uncertainty of a correct or incorrect behaviour is represented by the assumptions  $ok_1, ok_2, ok_3$  and  $ok_4$ . The next step is to describe the connections between the different components of the system. In our case of cascaded inverters each output of a gate is the input of the next gate, whereas the input of the first and the output of the last inverter represent the input and the output of the whole system:

$$in \leftrightarrow in_1, out_1 \leftrightarrow in_2, out_2 \leftrightarrow in_3, out_3 \leftrightarrow in_4, out_4 \leftrightarrow out. \tag{51}$$

To obtain a complete description of the model, we have to add the observed input and output values as facts to the knowledge base:

$$\neg in, out. \tag{52}$$

Finally, the model is given by a set  $A = \{ok_1, ok_2, ok_3, ok_4\}$  of assumptions, a set  $P = \{in, in_1, in_2, in_3, in_4, out_1, out_2, out_3, out_4, out\}$  of propositions and a set  $\Sigma$  of formulas in  $\mathcal{L}_{A+P}$ . The main computational task is then to determine the quasi-support of the contradiction  $Sp'(\perp)$ . In this example we obtain (conjunctions  $\wedge$  are written as multiplications):

$$\begin{aligned}
Sp'(\perp) &= ok_1 ok_2 ok_3 ok_4 \vee \neg ok_1 \neg ok_2 \neg ok_3 \neg ok_4 \vee \\
&ok_1 ok_2 \neg ok_3 \neg ok_4 \vee ok_1 \neg ok_2 ok_3 \neg ok_4 \vee \\
&\neg ok_1 ok_2 ok_3 \neg ok_4 \vee ok_1 \neg ok_2 \neg ok_3 ok_4 \vee \\
&\neg ok_1 ok_2 \neg ok_3 ok_4 \vee \neg ok_1 \neg ok_2 ok_3 ok_4.
\end{aligned} \tag{53}$$

This formula says that it is impossible to have exactly 0, 2 or 4 correctly working inverters, i.e. the number of faulty gates must be either 1 or 3. An explanation of the system behaviour in the sense of minimal sets of faulty components (Reiter, 1987) is given by the support of the tautology:

$$\begin{aligned}
Sp(\top) = \neg Sp'(\perp) &= ok_1 ok_2 ok_3 \neg ok_4 \vee ok_1 ok_2 \neg ok_3 ok_4 \vee \\
&ok_1 \neg ok_2 ok_3 ok_4 \vee \neg ok_1 ok_2 ok_3 ok_4 \vee \\
&ok_1 \neg ok_2 \neg ok_3 \neg ok_4 \vee \neg ok_1 \neg ok_2 ok_3 \neg ok_4 \vee \\
&\neg ok_1 ok_2 \neg ok_3 \neg ok_4 \vee \neg ok_1 \neg ok_2 \neg ok_3 ok_4. \quad (54)
\end{aligned}$$

Theorem 1 helps us now to determine the supports of more specific hypotheses. If for example we are interested in the arguments for components 1 and 2 both being intact, we obtain as support for the hypothesis  $ok_1 \wedge ok_2$  the formula:

$$\begin{aligned}
Sp(ok_1 \wedge ok_2) &= (ok_1 \wedge ok_2) \wedge \neg Sp'(\perp) \\
&= (ok_1 \wedge ok_2) \wedge Sp(\top) \\
&= ok_1 ok_2 \neg ok_3 ok_4 \vee ok_1 ok_2 ok_3 \neg ok_4. \quad (55)
\end{aligned}$$

Similarly, it is easy to determine the supports for all possible hypotheses  $h \in \mathcal{L}_A$ . This gives us different indications and possible explanations of the discrepancy between the observed and the correct system behaviour. In some cases this information can be used to decide whether a component should be replaced or not, but it is usually not sufficient. More information is obtained when probabilities are assigned to the assumptions and the corresponding degrees of support are computed. Because the example of cascaded inverters is a completely symmetric system, it is clear that if the probabilities of failures are the same for all components then it will not help much in finding the broken gate(s). By assuming the probabilities  $p_1 = p_2 = p_3 = p_4 = 0.8$  for the correct behaviour of the corresponding components, we get the same degrees of support:

$$sp(ok_1) = sp(ok_2) = sp(ok_3) = sp(ok_4) = 0.72. \quad (56)$$

In order to make the example more interesting, assume that faulty components always return 0. The system description has to be slightly modified :

$$\neg ok_1 \rightarrow \neg out_1 \quad \text{instead of} \quad \neg ok_1 \rightarrow (in_1 \leftrightarrow out_1),$$

$$\begin{aligned}
\neg ok_2 \rightarrow \neg out_2 & \text{ instead of } \neg ok_2 \rightarrow (in_2 \leftrightarrow out_2), \\
\neg ok_3 \rightarrow \neg out_3 & \text{ instead of } \neg ok_3 \rightarrow (in_3 \leftrightarrow out_3), \\
\neg ok_4 \rightarrow \neg out_4 & \text{ instead of } \neg ok_4 \rightarrow (in_4 \leftrightarrow out_4).
\end{aligned} \tag{57}$$

This modification leads to the following results:

$$Sp'(\perp) = ok_1 ok_3 \vee \neg ok_2 ok_3 \vee \neg ok_4, \tag{58}$$

$$Sp(\top) = \neg ok_1 ok_2 ok_4 \vee \neg ok_3 ok_4, \tag{59}$$

$$Sp(ok_1) = ok_1 \neg ok_3 ok_4, \tag{60}$$

$$Sp(ok_2) = \neg ok_1 ok_2 ok_4 \vee ok_2 \neg ok_3 ok_4, \tag{61}$$

$$Sp(ok_3) = \neg ok_1 ok_2 ok_3 ok_4, \tag{62}$$

$$Sp(ok_4) = \neg ok_1 ok_2 ok_4 \vee \neg ok_3 ok_4. \tag{63}$$

These results show that our model is no longer symmetric. To find the best component to replace we assign again the same probabilities  $p_1 = p_2 = p_3 = p_4 = 0.8$  to the assumptions  $ok_1, \dots, ok_4$ , and we get the following degrees of support:

$$sp(ok_1) = 0.49, sp(ok_2) = 0.88, sp(ok_3) = 0.39, sp(ok_4) = 1.0. \tag{64}$$

The last inverter does certainly work correctly. All other inverters could possibly be broken, but out of these the third one has the lowest degree of support of working properly. This is a fairly strong indication to replace the third inverter.

## 4.2 Binary Adder

Another instance of a model-based diagnostic problem which we place into the framework of assumption-based modelling is the binary adder example. Figure 3 shows a logical circuit corresponding to a binary adder:  $in_1$  and  $in_2$  are the two bits to be added,  $in_3$  is the carry bit from a previous addition,  $out_1$  is the sum of the three bits and  $out_2$  is the carry bit of this sum. The components of this circuit are logical and-gates ( $A_1, A_2$ ), exclusive or-gates ( $X_1, X_2$ ) and an ordinary or-gate ( $O_1$ ). An and-gate is a device whose output is the logical conjunction of its two binary inputs, an exclusive or-gate produces 1 if exactly one of its two inputs equals 1, and an or-gate has as its

output the logical disjunction of its two binary inputs. If the inputs  $in_1 = 1$ ,  $in_2 = 0$ ,  $in_3 = 1$  and the outputs  $out_1 = 1$  and  $out_2 = 0$  are observed, then these observations are clearly in contradiction with the expected behaviour of this system, namely  $out_1 = 0$  and  $out_2 = 1$ . Therefore, some gates are not working properly and the question is which ones.

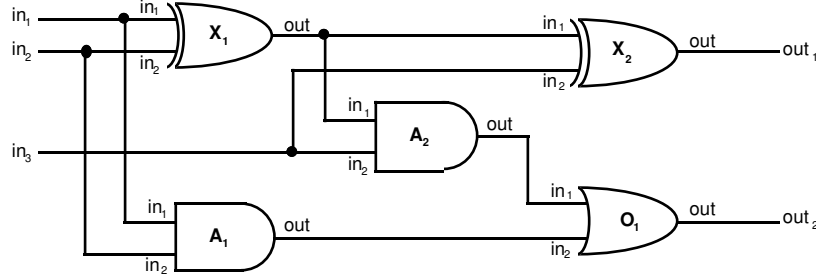


Figure 3: The logical circuit of a binary adder.

The correct behaviour of these five components can be modeled like the inverter components in the previous example: (the symbol  $\oplus$  represents the exclusive or)

$$\begin{aligned}
ok(A_1) &\rightarrow (out(A_1) \leftrightarrow in_1(A_1) \wedge in_2(A_1)), \\
ok(A_2) &\rightarrow (out(A_2) \leftrightarrow in_1(A_2) \wedge in_2(A_2)), \\
ok(X_1) &\rightarrow (out(X_1) \leftrightarrow in_1(X_1) \oplus in_2(X_1)), \\
ok(X_2) &\rightarrow (out(X_2) \leftrightarrow in_1(X_2) \oplus in_2(X_2)), \\
ok(O_1) &\rightarrow (out(O_1) \leftrightarrow in_1(O_1) \vee in_2(O_1)).
\end{aligned} \tag{65}$$

We assume different kinds of failure: broken and-gates produce precisely the opposite of what they should, broken or-gates always return 0, and for broken exclusive or-gates, every result is possible (even the correct one):

$$\begin{aligned}
\neg ok(A_1) &\rightarrow (out(A_1) \leftrightarrow \neg in_1(A_1) \vee \neg in_2(A_1)), \\
\neg ok(A_2) &\rightarrow (out(A_2) \leftrightarrow \neg in_1(A_2) \vee \neg in_2(A_2)), \\
\neg ok(X_1) &\rightarrow \top, \\
\neg ok(X_2) &\rightarrow \top, \\
\neg ok(O_1) &\rightarrow \neg out(O_1).
\end{aligned} \tag{66}$$

The inputs of the gates  $X_1$ ,  $A_1$ ,  $A_2$  correspond to the inputs  $in_1$ ,  $in_2$ ,  $in_3$  of the binary adder, and the outputs of  $X_2$  and  $O_1$  correspond to  $out_1$  and  $out_2$ .

$$\begin{aligned}
in_1 &\leftrightarrow in_1(X_1) \leftrightarrow in_1(A_1), \\
in_2 &\leftrightarrow in_2(X_1) \leftrightarrow in_2(A_1), \\
in_3 &\leftrightarrow in_2(A_2) \leftrightarrow in_2(X_2), \\
out_1 &\leftrightarrow out(X_2), \\
out_2 &\leftrightarrow out(O_1).
\end{aligned} \tag{67}$$

According to Figure 3, the internal connections between the components can be written as

$$\begin{aligned}
out(X_1) &\leftrightarrow in_1(X_2) \leftrightarrow in_1(A_2), \\
out(A_2) &\leftrightarrow in_1(O_1), \quad out(A_1) \leftrightarrow in_2(O_1),
\end{aligned} \tag{68}$$

and we have 5 observed input and output values

$$in_1, \neg in_2, in_3, out_1, \neg out_1. \tag{69}$$

This is the complete model description. It consists of the set of assumptions

$$A = \{ok(A_1), ok(A_2), ok(X_1), ok(X_2), ok(O_1)\}, \tag{70}$$

the set of propositional symbols  $P = \{in_1, in_2, \dots\}$  and a set  $\Sigma$  of formulas in  $\mathcal{L}_{A+P}$ . The next step is to compute the quasi-support of the contradiction:

$$\begin{aligned}
Sp'(\perp) &= ok(X_1)ok(X_2) \vee \\
&\quad ok(O_1)\neg ok(A_1) \vee \\
&\quad ok(O_1)ok(X_1)ok(A_2) \vee \\
&\quad ok(O_1)ok(X_2)\neg ok(A_2).
\end{aligned} \tag{71}$$

Now we know that the exclusive or-gates cannot both work properly and it is not possible that the or-gate works correctly while the first and-gate is broken, etc. In order to get a more precise diagnosis, consider the support of the tautology:

$$\begin{aligned}
Sp(\top) &= \neg ok(O_1)\neg ok(X_1) \vee \\
&\quad \neg ok(O_1)\neg ok(X_2) \vee \\
&\quad ok(A_1)ok(A_2)\neg ok(X_1) \vee \\
&\quad ok(A_1)\neg ok(A_2)\neg ok(X_2) \vee \\
&\quad ok(A_1)\neg ok(X_1)\neg ok(X_2).
\end{aligned} \tag{72}$$

These conjunctions are the possible explanations of the system behaviour. Since all conjunctions contain either  $ok(X_1)$  or  $ok(X_2)$  as negated literals, we know that at least one of the exclusive or-gates must be broken, but the question is which one ? In order to have more information we introduce the probability  $p = 0.9$  for the correct behaviour of all five components, i.e.

$$\begin{aligned} P(ok(A_1)) &= P(ok(A_2)) = P(ok(X_1)) \\ &= P(ok(X_2)) = P(ok(O_1)) = 0.9. \end{aligned} \quad (73)$$

This enables us to determine the numerical degrees of support:

$$\begin{aligned} sp(ok(A_1)) &= 0.98, \\ sp(ok(A_2)) &= 0.9, \\ sp(ok(X_1)) &= 0.16, \\ sp(ok(X_2)) &= 0.75, \\ sp(ok(O_1)) &= 0.81. \end{aligned} \quad (74)$$

The first exclusive or-gate  $X_1$  has the lowest degree of support of working properly. This is a strong indication to replace this component first. If the replacement of  $X_1$  does not repair the system, e.g. if still the same input and output values are observed, then we know that there must be other broken components. To find them the fact  $ok(X_1)$  is added to the knowledge base (assuming that replaced components are always working correctly) and redo the computation. This produces the following explanations:

$$\begin{aligned} Sp(\top) &= ok(X_1) \neg ok(O_1) \neg ok(X_2) \vee \\ &ok(X_1) ok(A_1) \neg ok(A_2) \neg ok(X_2). \end{aligned} \quad (75)$$

In both cases  $X_1$  works correctly and at least two components ( $X_2$  and either  $A_1$  or  $O_1$ ) are broken. The question is whether to replace  $A_1$  or  $O_1$ .

$$\begin{aligned} sp(ok(A_1)) &= 0.95, \\ sp(ok(A_2)) &= 0.5, \\ sp(ok(X_1)) &= 1.0, \\ sp(ok(X_2)) &= 0.0, \\ sp(ok(O_1)) &= 0.45. \end{aligned} \quad (76)$$

The or-gate  $O_1$  has a lower degree of support than  $A_1$  to work properly and should be replaced first. If after the replacement of  $O_1$  the same observed values remain, we can add the fact  $ok(O_1)$  to the knowledge base and redo the computations. Then we obtain a precise diagnosis:

$$Sp(\top) = ok(X_1)ok(A_1)ok(O_1)\neg ok(A_2)\neg ok(X_2), \quad (77)$$

i.e.  $X_2$  and  $A_2$  are broken,  $X_1$ ,  $A_1$  and  $O_1$  are intact.

## References

- Abraham, J.A. 1979. An Improved Algorithm for Network Reliability. *IEEE Transactions on Reliability*, **28**, 58–61.
- Heidtmann, K.D. 1989. Smaller Sums of Disjoint Products by Subproduct Inversion. *IEEE Transactions on Reliability*, **38**(3), 305–311.
- Inoue, K. 1992. Linear resolution for consequence finding. *Artificial Intelligence, Elsevier Science Publisher B.V. (Amsterdam)*, **56**, 301–353.
- Kohlas, J. 1993a. *Symbolic Evidence, Arguments, Supports and Valuation Networks*. Tech. Rep. 93-03. Institute of Informatics, University of Fribourg.
- Kohlas, J. 1993b. Symbolic Evidence, Arguments, Supports and Valuation Networks. *Pages 186–198 of: M. Clarke, M. Kruse, & Moral, S. (eds), Symbolic and Quantitative Approaches to Reasoning and Uncertainty*. Springer.
- Kohlas, J. 1994. *Mathematical Foundations of Evidence Theory*. Tech. Rep. 94-09. Institute of Informatics, University of Fribourg.
- Kohlas, J., & Monney, P.A. 1994. *Probabilistic Assumption-Based Reasoning*. Tech. Rep. 94-22. Institute of Informatics, University of Fribourg.
- Reiter, R. 1987. A Theory of Diagnosis From First Principles. *Artificial Intelligence, Elsevier Science Publisher B.V. (Amsterdam)*, **32**, 57–95.

- Reiter, R., & de Kleer, J. 1987. Foundations of Assumption-based Truth Maintenance Systems. *Proc. Amer. Assoc. A.I.*, 183–188.
- Shenoy, P. 1994. Using Dempster-Shafer’s Belief Function Theory in Expert Systems. *Pages 395–414 of: Yager, R.R., Kacprzyk, J., & Fedrizzi, M. (eds), Advances The Dempster-Shafer Theory of Evidence.* Wiley.
- Siegel, P. 1987. *Représentation et Utilisation de la Connaissance en Calcul Propositionnel.* Ph.D. thesis, Université d’Aix-Marseille II. Luminy, France.