Propositional Non Clausal Deduction and Diagnosis*

Anavai Ramesh   Neil V. Murray
Institute for Programming & Logics
Department of Computer Science
University at Albany, Albany, NY 12222
{rameshag,nvm}@cs.albany.edu

Bernhard Beckert   Reiner Hähnle
Institute for Logic, Complexity,
and Deduction Systems
University of Karlsruhe, D-76128 Karlsruhe
⟨name⟩@ira.uka.de
http://i12www.ira.uka.de/~⟨name⟩/

In [11] Reiter laid the foundations of the formal theory of an approach to diagnosis known as *diagnosis from first principles*. His work was based on the work of many researchers, notably that of de Kleer [4] and Genesereth [5]. In diagnosis from first principles, we have a logic based description of some system (e.g., a circuit) and an observation of the system's behavior. We then try to find a set of *components* in the system which, when assumed to be *abnormal*, explains the discrepancy between the intended behavior of the system and the observation. Usually the number of diagnoses will be exponential in the total number of components. Reiter uses *minimal diagnosis* to characterize all diagnoses, and gives a procedure (without an implementation) which can compute all minimal diagnoses. de Kleer [2] uses a less formal method, which requires the failure probabilities of the components to be known, in his diagnosis system. Mozetič and Holzbaur [6] describe an algorithm which uses Prolog for specifying the system description. Their technique is a purely symbolic method based on Reiter's theory. However, all these techniques require the system description to be in conjunctive normal form (CNF) or some form which is close to CNF. Therefore, any formula not in CNF must be converted to CNF before applying the diagnosis system. Relying on CNF or any other clause form may cause an exponential blow-up even before the diagnosis algorithms can be applied. Efficient clause form translations [8] commonly used in theorem provers cannot be used here because they do not preserve equivalence during transformation. The diagnosis systems of Reiter [11], de Kleer [2], and of Mozetič and Holzbaur [6] also require the generation of *minimal conflicts*. Generating minimal conflicts causes an additional blow up.

In [10] we describe a new technique for computing minimal diagnoses of a system based on Reiter's theory; also, modifications are introduced so that only *single fault* diagnoses are generated. This approach does not rely on a clause form representation (although it is applicable to systems represented in clause form), nor does it require generating the set of minimal conflicts. The approach is based on *dissolution*, an inference rule described in [7] for formulas in negation normal form (NNF). We have obtained experimental results on the performance of our techniques on commonly used benchmark problems [10].

A *system* is a pair $(SD, Comp)$ where SD, the *system description*, is a propositional formula, and $Comp$, the set of *system components*, is a finite set. For each component c we have a propositional variable denoted by $ab(c)$, that is interpreted to mean that component c is abnormal. We use AbV to denote the set of these variables. An *observation* Obs is a propositional formula. A *diagnosis* for the system $(SD, Comp)$ with observation Obs is a set $\Delta \subset AbV$ such that the propositional formula

$$\mathcal{F}(SD, Obs, \Delta) \;=\; SD \wedge Obs \wedge \left(\bigwedge_{ab(c)\in\Delta} ab(c) \right) \wedge \left(\bigwedge_{ab(c)\in(AbV\setminus\Delta)} \overline{ab(c)} \right)$$

is consistent. A diagnosis Δ is *minimal* if no proper subset of Δ is a diagnosis. A diagnosis Δ is a *single fault* diagnosis if Δ is singleton set, otherwise it is a *multiple fault* diagnosis.[1]

Dissolution [7] is an inference rule that preserves equivalence and that terminates naturally in the propositional case. If we dissolve in formula F until it is linkless, the resulting formula is called the *full dissolvent* of F; we denote it by $FD(F)$. The set of conjunctive paths[2] (c-paths) in $FD(F)$ is unique: The satisfiable c-paths in F.

In [9] we describe an algorithm PI which can compute the following set $\pi(F)$, under the assumption that the formula F has no unsatisfiable c-paths ($\ell(p)$ denotes the set of literals of a c-path p):

$$\pi(F) = \{\ell(p) \;:\; p \text{ is a c-path through } F\text{, and for all c-paths } q \text{ through } F\text{: } \ell(q) \not\subset \ell(p)\}$$

Theorem 1 *Let $(SD, Comp)$ be a system with observation Obs, and let $\mathcal{D}$ be the set*

$$\{\ell(p) \cap AbV \;:\; p \textit{ is a c-path in } FD(Obs \wedge SD)\} \;.$$

Then $\pi(\mathcal{D})$ contains all the minimal diagnoses of $(SD, Comp)$ with observation Obs.[3]

*This research was supported in part by National Science Foundation Grant CCR-9101208.

[1] These definitions are a propositional version of those used by Reiter [11].

[2] A conjunctive path in a formula F is a conjunction of literals that is a clause in the disjunctive normal form (DNF) of F.

[3] Note that we are treating $\mathcal{D}$ as a DNF formula when applying π to it.

We may compute all minimal diagnoses by restricting each set to literals containing only positive variables from AbV while computing $\pi(FD(SD \wedge Obs))$. In fact, our implementation uses a further refinement based on pure literals[4]:

Theorem 2 *Let K be a pure literal in the formula F, where $K \notin AbV$, and let*

$$\mathcal{D} = \{\ell(p) \cap AbV \ : \ p \text{ is a satisfiable c-path in } F\}$$
$$\mathcal{D}' = \{\ell(p) \cap AbV \ : \ p \text{ is a satisfiable c-path in } DC(K,G)\}$$

Then $\pi(\mathcal{D}) = \pi(\mathcal{D}')$.

Here, $DC(K,G)$ denotes the *d-path complement* of K in F.[5] The DC operator strictly reduces the size of the formula on which it is applied. We apply this reduction whenever we discover a pure occurrence of a literal. Therefore, by applying this theorem during the process of dissolving, we can significantly reduce the combinatorial explosion. By definition, all the literal occurrences in a full dissolvent are pure. Therefore, at the end of the process we will be left with a formula which has literals from the set AbV only. Thus we may use PI to compute the set $\pi(FD(SD \wedge Obs))$.

To find all and only single faults, we restructure the formula during the process of dissolving so as to eliminate c-paths which have at least two different variables from the set AbV occurring in them. By eliminating such paths, potential multiple faults are also eliminated. These operations strictly reduce the size of the formula and therefore improve the performance of the algorithm.

We have currently implemented both the single fault and multiple fault diagnosis algorithms. Our implementation is written in C/C++ (running on a Sun Sparc 5), and has been enhanced with the *anti-link* operations from [1] to reduce the number of subsumption checks, and uses a *trie*-like data structure (described in [3]). We have used the familiar n-bit carry adder example from [11]. The n-bit adder circuit is considered to be a difficult example for diagnosis systems. The following table gives running times (in seconds) for various observations of the system for the adder example. Each observation was either randomly chosen or was the specific observation where all the inputs are 0 and all the outputs except the n^{th} bit are 0, yielding exactly $3n - 1$ minimal diagnoses.

Problem	# Gates	Mozetič & Holzbaur	Our System	# Diagnoses	Observation
3-bit adder	15	260.47	0.41	204	random
3-bit adder	15	0.01	0.35	2	random
3-bit adder	15	0.13	0.47	8	specific
6-bit adder	30	$>$ 2 hours	524.26	17	specific

We also compared our system with that of Mozetič and Holzbaur when computing single faults only. We employed the specific observation used in the above table. To produce single faults only, the input to the Mozetič-Holzbaur algorithm was appropriately augmented. It was much slower than ours. (We note that perhaps the Mozetič-Holzbaur algorithm itself could be modified to handle single faults, as opposed to modifying the input, although we see no obvious way to do so.)

References

[1] B. Beckert, R. Hähnle, N. Murray, and A. Ramesh. On anti-links. In *Proc., 5th Int. Conf. on Logic Programming and Automated Reasoning (LPAR), Kiev,* LNCS 882, pages 275–289. Springer, 1994.

[2] J. de Kleer. Focusing on probable diagnosis. *Proc. of AAAI-91, Anaheim, CA,* pages 842–848, 1991.

[3] J. de Kleer. An improved incremental algorithm for computing prime implicants. *Proceedings of AAAI-92, San Jose, CA*, pages 780-785, 1992.

[4] J. de Kleer and B. Williams. Diagnosing multiple faults, *Artificial Intelligence*, 56:197–222, 1987.

[5] M. R. Genesereth. The use of design descriptions in automated diagnosis. *Artificial Intelligence*, 24:411–436, 1984.

[6] I. Mozetič and C. Holzbaur. Controlling the complexity in model-based diagnosis. To appear in *Annals of Mathematics and Artificial Intelligence*. Preliminary version in *Proceedings of the European Conference on Artificial Intelligence*, pages 729–733, John Wiley & Sons, 1992.

[7] N. Murray and E. Rosenthal. Dissolution: Making paths vanish. *Journal of the ACM*, 40(3):504–535, 1993.

[8] D. Plaisted and S. Greenbaum. A structure preserving clause form translation. *Journal of Symbolic Computation*, 2:293–304, 1986.

[9] A. Ramesh, G. Becker, and N. Murray. CNF and DNF considered harmful for computing prime implicants/implicates. To appear in *Methods of Logic in Computer Science*, 1995.

[10] A. Ramesh and N. Murray. An application of non clausal deduction in diagnosis. Technical Report TR 95-1, Dept. of Computer Sci., University at Albany, January 1995.

[11] R. Reiter. A theory of diagnosis from first principles. *Artificial Intelligence*, 32:57–95, 1987.

[4]A literal K is *pure* in a formula F, if there is no c-path in F containing both K and its negation.

[5]The *d-path complement* of K in F is a formula containing the disjunctive paths (d-paths) of F that do not go through K (see [7]).