DECOMPOSITION INTO INDEPENDENT DIAGNOSIS SUBPROBLEMS

Yury Tsybenko

Intelligent Tutoring Systems Laboratory
Glushkov Institute for Cybernetics
40, Prospect Akademica Glushkova, 252022, Kiev, Ukraine
Phone: (044)-2676-934
E-mail: tsybenko@its.freenet.kiev.ua

Abstract. A decomposition of a complex problem into independent subproblems is a standard method to tackle large AI problems. This paper studies a notion of diagnosis independence related to model-based diagnosis and proposes a concept of P-decomposition of a diagnosis problem into independent diagnosis subproblems (IDS). We show that a system P-decomposition depends critically on the system structure and the measurements performed in the system. Taking into account both these factors we introduce a notion of a conflict graph and show that a system decomposition into IDS may be reduced to the system conflict graph decomposition.

This paper develops a formal algorithm for localization of multiple faults based on P-decomposition. We show that a strategy of measurement selection for P-decomposition may be efficiently guided by pure "the first principle information" such as current observation and the system topology. In many cases the algorithm runs in logarithmical time complexity compared to general diagnosis time.

1 Introduction

Model-based diagnosis (MBD) is a very dynamic and a wide field of research based on standard AI technique such as predicate logic, heuristic search and qualitative simulation. Current research in this area concerns a wide class of systems including physical and medical systems, student models, etc. Several practical applications, such as electronic circuits, energy transport networks etc., have shown the industrial potential of the approach.

In the recent years many new important results have been developed in the field of MBD. Among these are: multiple behavioral modes [de Kleer and Williams, 1989], focusing on probable diagnoses [de Kleer, 1991], physical negation [Struss and Dressler, 1989] and physical impossibility [Friedrich et al., 1990], focusing on independent diagnosis problems [Freitag and Friedrich, 1992] and hierarchy design [Hamscher, 1990].

However, the difficulties for building large diagnostic systems remains. One of the main obstacles for building large diagnostic systems is the great number of possible diagnoses needed to be considered. Many real world applications deal with the system composed of thousands of components. As examples let us consider energy transport network and telecommunication network. As these systems are very large, diagnosis in such systems is a complex time consuming task and may cause high breakdown costs. Practically such systems cannot wait until whole system will be diagnosed.

A standard approach to tackle a large AI problem is to decompose it into subproblems. Actually, the cardinality of the diagnosis set grows exponentially in the number of system components (system dimension). Reducing the system dimension we exponentially reduce the search space for diagnosis. Let us assume that suspected components are located within a certain subsystem. Is it possible to perform diagnosis only for suspected subsystem instead of performing the whole system diagnosis? The lack of logically sound methods, except hierarchy design, which allow to reduce diagnosis of the overall system to the diagnosis of its subsystems is a major constraint on a wider applicability of such an approach.

This paper studies a concept of diagnosis independence and proposes a method of decomposition of a diagnosis problem into the independent diagnosis subproblems (IDS). We show that there is a certain set of measurements that, when performed, reduces the diagnosis of the overall system to the diagnosis of its subsystems. We prove that the total diagnosis set is decomposed into the Cartesian product of correspondent local diagnosis sets. Proposed method does not rely on a hierarchy structure of a system to be diagnosed. In contrast to hierarchy decomposition [Genesereth, 1984], [Hamscher, 1990] this method is called P(product)-decomposition.

In this paper we study the measurements that enable P-decomposition. Traditional GDE [de Kleer, Williams 1987] approach applies probabilistic analysis to decide what measurement to take next. This paper proposes a strategy of measurement selection which is efficiently guided by the system description and current observation. To develop this strategy we introduce a notion of a conflict graph. It has been shown that P-decomposition of a system may be reduced to the system conflict graph decomposition. A new interesting strategy for next measurement selection based on a conflict graph decomposition has been developed.

2 Motivation

This section presents two examples illustrating the crucial points of our research. As the first example we use n=2k cascaded inverters.

Example 1. Suppose that the device input is 1 and the output is 0 violating the prediction and telling us that at least one of the components is faulted. So additional measurements are necessary to locate the faults. If all gates fail with equal likelihood a human diagnostics should choose the inverter *Ik* output as the best next measurement. Why measuring *Ik* output is chosen as the best diagnostic action instead of measuring another device point? What technique is the base for such an intuitions?

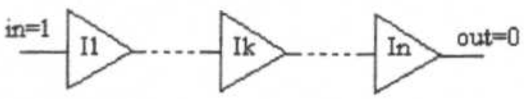

Figure 1. n=2k cascaded inverters.

Measuring the component *Ij*, *(j=1,n-1)* output, no matter what the value has been obtained as the result of this measurement, divides the given system into two subsystems. Actually, component *Ij* output is the output of the subsystem composed of the components *I1,...,Ij*, and is the input of the subsystem composed of the components *Ij+1,...,In*. As all inputs and outputs of both subsystems are known the diagnosis of each subsystem is independent of the overall system diagnosis. In our example measuring *Ik* provides the best (into equal parts) decomposition of the given diagnosis problem into independent diagnosis subproblems.

At the first example we use the measurements to decompose a system into independent diagnosis subproblems. The second example illustrates another approach to determining the IDS. It shows that there is a situation when the IDS may be found without probing, solely on the base of current observation and the knowledge of component behavior.

Example 2. Consider the binary circuit depicted on Figure 2. If the input *in1(AND)* is 0 then the component *AND* behavior is independent of its second input *in2(AND)*. Moreover, in this case the diagnosis of the subsystem composed of the component *AND* and the *Device 2* is independent of the *Device 1* diagnosis.

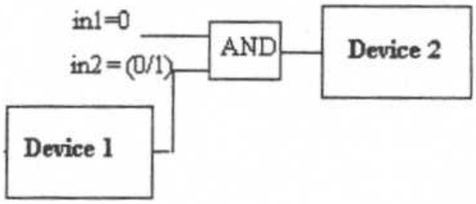

Figure 2. AND is *and* gate, and *Device1, Device2* are arbitrary circuits

What is nature of the IDS for arbitrary system? Are there common criterion for the IDS in above examples? Are there efficient strategy for building the IDS and how it may be applied to tackle complex diagnosis problems? The answer to these questions is the topic of the paper.

3 Preliminaries

This section follows the formal framework for model-based diagnosis [Reiter, 1987] and [de Kleer et al., 1992].

Definition 1. A system is a triple *(Sd,Comps,Obs)* where: *Comps*, the system components, is a finite set of constants; *Sd*, the system description, is a set of first order sentences, defining the system component behavior and their connection; *Obs*, a set of observations, is a set of first order sentences.

We adopt the Reiter's convention that *AB(c)* is a literal which holds when component $c \in Comps$ is abnormal. We use the modeling stance that a component is abnormal if something is wrong physically with it that may cause the incorrect output emergency. A model-based diagnosis for a system is a conjecture that states: a certain set of system components are abnormal, the other components are normal. This conjecture must satisfy the system description and explain the symptoms.

Definition 2. Given two sets of components C_p and C_n define $D(C_p, C_n)$ to be a conjunction:

$$[\wedge AB(c) : c \in C_p] \wedge [\wedge \neg AB(c) : c \in C_n].$$

A diagnosis for *(Sd,Comps,Obs)* is a formula $D(\Delta, Comps-\Delta)$ such that: $Sd \cup Obs \cup D(\Delta, Comps-\Delta)$ is satisfiable. A diagnosis $D(\Delta, Comps-\Delta)$ is a minimal diagnosis iff for no proper subset Δ_1 of Δ is $D(\Delta_1, Comps-\Delta_1)$ a diagnosis. We shall write Δ instead of $D(\Delta, Comps-\Delta)$, since subset Δ determines the diagnosis.

Definition 3. An *AB*-literal is *AB(c)* or $\neg AB(c)$ for some $c \in Comps$. An *AB*-clause is a clause consisting of *AB*-literals. A conflict of *(Sd, Comps, Obs)* is an *AB*-clause entailed by $Sd \cup Obs$. Conflict *C* is minimal if no proper subclause of *C* is a conflict for *(Sd, Comps, Obs)*.

4 Diagnosis independence

So far an intuitive concept of diagnosis independence has been considered. In this section we formalizes such an intuition. Let us consider a formal statement that the diagnosis of a subsystem is independent of the rest of a system. In this case the diagnosis of 'the rest of a system' is also independent of the subsystem, in other words the concept of diagnosis independence is symmetric. So, we have to consider a decomposition into independent diagnosis subproblems. What do we mean saying that "a diagnosis problem is decomposed into independent subproblems A and B"? It means that for each local diagnoses for the subsystem A its conjunction with a local diagnosis for the subsystem B is a diagnosis for the system, and vice versa for each system diagnosis its projections on the subsystems A or B are local diagnoses for the subsystems

4.1 Subsystems and local diagnoses

To introduce the formal definition of diagnosis independence let as consider the notions of a subsystem and local diagnosis.

Definition 4. Let *(Sd, Comps, Obs)* be a system. A subsystem is a triple *(Sd1, Comps1, Obs1)*, where *Comps1* is a subset of *Comps*. The subsystem description *Sd1* contains the axioms related to the components from *Comps1*, and the observation *Obs1* contains the observation results related to the components from *Comps1*. We shall write *Sub(C1,...,Ck)* to indicate a subsystem composed of components *C1,...,Ck*. Each diagnosis for a *(Sd1, Comps1, Obs1)* is called a local diagnosis for subsystem.

Lemma 1. Let *(Sd, Comps, Obs)* is a system, and Δ is a diagnosis. Δ projection on a subsystem is a local diagnosis for subsystem.

Proof. Let $\Delta 1$ be Δ projection on a subsystem *(Sd1, Comps1, Obs1)*. If Δ is a diagnosis, then $Sd \cup Obs \cup \Delta$ is satisfiable. Hence $Sd1 \cup Obs1 \cup \Delta 1$ is also satisfiable, where $Sd1 \subseteq Sd$, $Obs1 \subseteq Obs$, $\Delta 1$ is a subconjunct of Δ containing *AB*-literals from Δ related to the components mentioned in Comps1.

However, the contrary to lemma 1 proves to be wrong. The following example shows that there are local diagnosis such that their combination is not a diagnosis for the system.

Example 3. The familiar circuit (Figure 1) consists of three multipliers *M1,M2,M3* and two adders *A1,A2*.

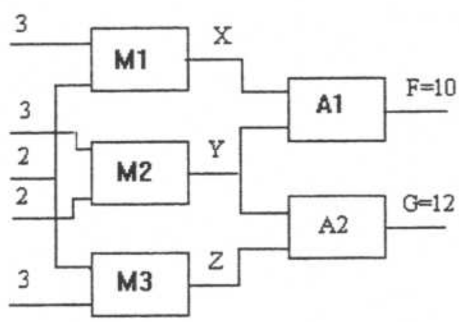

Figure 3. Familiar Circuit

The system description is given by:

$\text{ADDER}(x) \Rightarrow \{\neg AB(x) \Rightarrow Out(x)=In1(x)+In2(x)\}$
$\text{MULTIPLIER}(x) \Rightarrow \{\neg AB(x) \Rightarrow Out(x)=In1(x)*In2(x)\}$;

$Out(M1)=In1(A1)$, $Out(M2)=In2(A1)$,
$Out(M2)=In1(A2)$, $Out(M3)=In2(A2)$.

The subsystem *Subs(M2,M3)* description contains multiplier constraint, topology axioms: $Out(M2)=Y$, $Out(M3)=Z$, and a set of observation: $In2(M2)=In1(M3)=2$, $In1(M2)=In2(M3)=3$. Because the values of ports *Y, Z* are not known [$\varnothing$] is a diagnosis for *Subs(M2,M3)*; [$\varnothing$] is also a diagnosis for the subsystem Sub(A1,A2,M1). However, conjunction $[\neg AB(M2) \wedge \neg AB(M3)] \wedge [\neg AB(M1) \wedge \neg AB(A1) \wedge \neg AB(A2)]$ is not a diagnosis for the overall system.

4.2 Definition and criterion for diagnosis independence

Definition 5. A diagnosis problem *(Sd, Comps, Obs)* is decomposed into independent diagnosis subproblems (P-decomposed) iff total diagnosis set for *(Sd, Comps, Obs)* is the Cartesian product of correspondent subproblem diagnosis sets.

The notion of a conflict is a basic notion for model-based diagnosis. The conflicts represent the discrepancies between the predicted behavior and the observed behavior, and form the basis for the derivation of the diagnoses. The following theorem gives a criterion of P-decomposition in terms of conflicts.

Theorem 1. (*Criterion of P-decomposition*) A diagnosis problem *(Sd, Comps, Obs)* is decomposed into independent diagnosis subproblems *Sub(Comps1)* and *Sub(Comps-Comps1)* iff each conflict for *(Sd, Comps, Obs)* is located within one of the following subsystems *Sub(Comps1)*, *Sub(Comps-Comps1)*.

Proof. ($\Rightarrow$) Suppose *Sub(Comps1)* and *Sub(Comps-Comps1)* are independent diagnosis problems. The proof is by contradiction. Let us assume that there is the minimal conflict $C=L11 \vee ... \vee L1k \vee L21 \vee ... \vee L2j$, where $L11,...,L1k$ are *AB*-literals for components from *Comps1*, and $L21,...,L2j$ are *AB*-literals for components from *Comps-Comps1*. In this case there is a local diagnosis $\Delta 1$ for *Sub(Comps1)* such that $\Delta 1$ contains $L11 \wedge ... \wedge L1k$ as subconjunct. Assuming to the contrary we obtain that the clause $L11 \vee ... \vee L1k$ is a conflict which is a proper subclause of the minimal conflict *C*. In the similar way we obtain that there is the local diagnosis $\Delta 2$ for *Sub(Comps-Comps1)* containing $L21 \wedge ... \wedge L2j$ as subconjunct. However, the conjunction $\Delta 1 \wedge \Delta 2$ contains all *AB*-literals from the conflict *C* , and hence, is not a diagnosis for *(Sd,Comps,Obs)*

($\Leftarrow$) In accordance with [de Kleer et al., 1992] the conjunct $D(\Delta,Comps\text{-}\Delta)$ is a diagnosis for *(Sd,Comps,Obs)* if and only if it contains a kernel diagnosis as subconjunct. Hence, it is sufficient to prove the theorem for the set of kernel diagnoses. A simple algorithm of computing kernel diagnosis set from minimal conflict set has been proposed in [de Kleer et al., 1992]. Following this algorithm we obtain that a kernel diagnosis set for *(Sd,Comps,Obs)* is the Cartesian product of sets of kernel diagnoses for subsystems *Sub(Comps1)*, *Subs(Comps-Comps1)*.

Example 4 (Example 3 continued). Additional measurements are necessary for a decomposition into independent diagnosis subproblems. Measuring point *Y* provides *P*-decomposition of the given problem into three independent diagnosis subproblems *Sub(M1,A1)*, *Sub(M2)* and *Sub(M3,A2)*. Actually, when the value of point *Y* has been measured the correspondent I/O values for all above subsystems are known. So, the diagnosis of each subsystem may be performed independently.

5 Strategy of P-decomposition

There are two polar strategies for determining the IDS. As it has been shown the IDS may be either inferred on the base of the knowledge of component behavior and current observations (Example 2) or determined by performing additional measurements (Example 1). Does a general strategy for a decomposition into independent diagnosis subproblems exists? This section gives positive answers to these questions.

5.1 Conflict graph

In accordance with theorem 1 set of conflicts located within correspondent subsystems is necessary to decompose the given diagnosis problem into independent diagnosis subproblems. Let us assume that a component behavior is described by a set of constraints. Sets of constraints representing the different components are connected by shared variables. The actual observation is modeled as the value assignment to related variable.

Conflicts located within a certain subsystem are necessary for *P*-decomposition. A conflict geometry depends critically on the following parameters: 1) a system topology; 2) current observations. To develop a method of generating required conflicts which takes into account 1) and 2) we apply graphical representation of a system structure. Consider a graph with two types of nodes: the set of main nodes *Comps* represents the set of system components, the set of auxiliary nodes *Var* represents the set of shared variables, and the set *Sd* of graph edges represents component-variable connections. We shall call the graph *G* a structural graph for the given system. A conflict graph for *(Sd, Comps, Obs)* is a minimal subgraph *G(Obs)* of *G* containing those paths from *G* along which minimal conflicts for *(Sd, Comps, Obs)* have been derived. We assume that a conflict path begins at a main node and ends at a main node. Because the minimal conflict set depend on observations performed in a system, the conflict graph *G(Obs)* represents the current state of system observations.

5.2 Graphical criterion of P-decomposition

Let Comps1 is a subset of components, define *Suspect(Comps1)* is a subset of *Comps1* consisting of those components which are mentioned in minimal conflicts for *(Sd, Comps, Obs)*. The following theorem is a graphical analog of theorem1.

Theorem 2. (*Graphical criterion of P-decomposition*). A diagnosis problem *(Sd, Comps, Obs)* is decomposed into independent diagnosis subproblems *Sub(Comps1)* and *Sub(Comps-Comps1)* iff the sets of nodes *Suspect(Comps1)* and *Suspect(Comps-Comps1)* belong to different components (maximal connected subgraphs) of the conflict graph *G(Obs)*.

Proof. The proof follows from theorem 1.

5.3 Conflict graph and observations

As we have seen in example 3 additional measurements are necessary to perform *P*-decomposition. The measurements affect current set of minimal conflicts of a system being diagnosed. Consider a malfunctioning system composed of many components. The initial observation usually provides a few minimal conflicts covering almost all of the system components. Minimal conflicts provided by new measurements as a rule are proper subclauses of the early conflicts. Additional measurements progressively reduce the size of minimal conflicts, until faulty components have been located.

This section investigates how the measurements performed in a system affect the structure of the system conflict graph.

Example 5. Consider a circuit from example 1. The initial observation provides single minimal conflict including all system components *C1,...,Cn*. Correspondent conflict graph depicted on Figure 4 is a line graph composed of 2n nodes. Measuring the component *Ck* output provides new minimal conflict which is located within either *Sub(C1,...,Ck)* or *Sub(Ck+1,...,Cn)*. So, the resulting conflict graph may be reduced considerably.

In general, we cannot predict the structure of resulting conflict graph until a measurement outcome is not known. However, there is a conflict graph invariant which does not depend on a measurement outcome. The following theorems characterize this important property of a conflict graph.

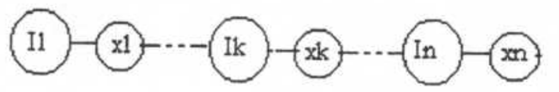

Figure 4. Conflict graph for cascaded inverters.

Theorem 3. Let *G(Obs)* is a conflict graph for *(Sd, Comps, Obs)*, and the auxiliary node *x* represents the variable *x*. Suppose that the value of the variable *x* is known. Then the conflict graph *G(Obs)* does not contain any edge connected to the node *x*.

Proof. The proof is by contradiction. Assuming to the contrary we obtain the minimal conflict *C* for *(Sd,Comps,Obs)* and at least the two components *ci, cj* mentioned in *C* which share the variable *x*. A conflict may be derived in the following way: starting from the known values the inference procedure propagates the known values through the system component constraints until the inconsistency has been detected. A minimal conflict is a minimal supporting environment for the inconsistency detected in the system [de Kleer, Williams 87].

To prove the theorem we have to build the minimal conflict *C1* such that *C1* subsumes *C* and correspondent path for the conflict *C1* derivation does not contain the variable *x*. To identify the required conflict we begin our search at the empty environment *E1*, adding to it the components in the same way that the conflict *C* was derived until the variable *x* has been met. Because the value of the variable *x* has been measured we can check the consistency of *E1*. If the observed value for *x* differs from the predicted value then the inconsistent environment *E1* represents the required conflict. Otherwise, we can refine the environment *E1* from those assumptions (components) that support the value assigned to the variable *x*. In this case at least one of the components *ci, cj* must be deleted from *E1*, contradicting to the fact that the conflict graph contains a path between *ci, cj*.

Theorem 4. Let *G(Obs)* is a conflict graph for *(Sd, Comps, Obs)*, and the auxiliary node *x* represents the variable *x*. Suppose that the component *c* behavior does not depend on the value assigned to the variable *x*. Then the conflict graph *G(Obs)* does not contain the edge joining the node *c* with the auxiliary node *x*.

The proof of theorem 4 is similar to the proof of theorem 3.

5.4 Selecting measurements for P-decomposition

To develop the strategy of measurement selection for *P*-decomposition we incorporate the measurements in a 'conflict graph formalism': measuring the variable *x* within a system entails removing all edges joining the variable *x* with the other conflict graph nodes from structural graph, and vice versa removing the edges connected to the variable *x* points out that the variable *x* must be measured.

Suppose we have to decompose the given problem *(Sd, Comps, Obs)* into two IDS *Sub(Comps1)* and *Sub(Comps-Comps1)*. To determine the measurements that enable P-decomposition we use the following Algorithm 1.

Algorithm 1.

Define *G* is a system structural graph, *BorderVar* is a set of border variables for *Sub(Comps1)* and *Sub(Comps-Comps1)*.

1. Loop until *BorderVar* is not empty:

2. For each $x \in BorderVar$ DO:
if the component *c* behavior does not depend on the value assigned to the variable *x* remove the edges joining the variable *x* with this component; generate new *BorderVar*.

3. If *BorderVar* is not empty DO:
select one variable $x \in BorderVar$ and measure *x*; delete all edges connected to the variable *x*; generate new *BorderVar*; go to 2.
END.

5.5 Algorithm of P-decomposition for localization of multiple faults

The results obtained in the previous section provide a formal background for a method for faulty component localization. The method is sequential decomposition of suspected parts of a system into IDS and looks as follows:

Algorithm 2 *(Informal algorithm of sequential P-decomposition).*

Define *SysList* is a list of subsystems.

1. *SysList =(Sd, Comps, Obs)*.
2. Loop until *SysList* is not empty.

3. For every subsystem *S* from *SysList* DO:

4. Applying Algorithm 1 decompose the subsystem *S* into IDS. If any discriminatory information is not available the best *P*-decomposition is provided by a half split.

5. For every new subsystem *S1* DO:
if the subsystem *S1* consists of only one node correspondent faulty component is isolated, otherwise add this subsystem to the *SysList*.
END

Remark 1. If for each system component the variable connected to its input may be measured Algorithm 2 gives single solution.

Remark 2. Each step of the algorithm prunes exponentially the diagnosis set. However, actual conflicts as well as actual diagnoses [Tsybenko, 94] are not lost.

Example 6. Consider the conflict graph for the system depicted on Figure 4. Because removing a variable node from this graph leads to the graph decomposition, each measurement provides *P*-decomposition of the given system. The best measurement is *xk*, because this measurement divides the given system into equal parts. Let us assume that measuring the component *Ck* output provides new minimal conflict which is located within the subsystem *Sub(C1,...,Ck)*. In this case the subsystem *Sub(Ck+1,...,Cn)* does not display the misbehavior so, it may be ignored. To locate the actual fault we have to repeat P-decomposition into equal parts for the resulting conflict graph until one element conflict has been located. A half split strategy requires logarithmic time complexity of general diagnosis time for single fault localization.

5.6 Single measurement P-decomposition

Practical diagnosis requires the discovery of faulty components in a minimum number of measurements. An important problem is to describe the case when single measurement provides *P*-decomposition. The following theorems 5, 6 proven in [Tsybenko, in prep.] present a solution of this problem.

If any discriminatory information is not available a half split strategy (illustrated in example 1) may be used in algorithm 1 for the next measurement selection. Theorem 5 generalizes the technique demonstrated in Example 1.

Theorem 5. Let *(Sd,Comps,Obs)* is a system, and *G* is the system structural graph. Suppose that the graph *G* is acyclic. Then each single measurement provides *P*-decomposition of the given system. Half split strategy of measurement selection requires logarithmical time complexity for localization of multiple faults.

Even if the structure of a conflict graph is very complex the single measurement *P*-decomposition exists. However, in this case algorithm 1 may run in non-logarithmic time. The following result holds in the case of arbitrary conflict graph.

Theorem 6. Let *(Sd,Comps,Obs)* is a system. Suppose that the variables connected to each component output can be measured. Then there is a sequential *P*-decomposition locating the set of faulty components such that each step of the *P*-decomposition requires only one measurement.

5.7 Incorporating a preference criterion in the algorithm of P-decomposition.

Diagnosis of complex devices composed of many components is difficult in part because the number of possible diagnoses grows exponentially in the number of system components. It is unacceptable to consider the set of all diagnoses. So, many diagnosis approaches apply a preference criterion to restrict the set of diagnoses needed to be considered. The preference criterion may be the number of suspects in a diagnosis [Freitag and Friedrich, 1992] or the probability of a diagnosis [de Kleer, 1991]. A preference criterion allows to focus the diagnosis process on certain parts of a system.

The effectiveness of Algorithm 2 improves considerably if a preference criterion is taken into account. In this case each step of the algorithm consists of two stages. At the first stage a preference criterion is applied to identify the suspected parts of a system. At the second stage P-decomposition is applied to separate these parts from the overall system. If the other parts of the system do not display the misbehavior, they may be ignored. To illustrate combined approach let us consider the following example taken from [de Kleer, 1991]. Probability of a diagnosis is used as a preference criterion.

Example 7. Consider *n*-bit adder *(b1,...,bn)* depicted on Figure 5. Suppose that all inputs are 0 and output of the *n-th* bit *Qn* is 1. Suppose that all gates fail with equal probability. Focused GDE locates probable diagnoses within the subsystem *Subs(bn,bn-1)*, there are only five probable diagnoses (called leading diagnoses): [S1*(bn.X1)*], [S1*(bn.X2)*], [S1*(bn-1A1)*], [S1*(bn-1O1)*], [S1*(bn-1A2)*], where [S_1*(X)*] indicates the candidate in which component *X* output is in mode output-stuck-at-1.

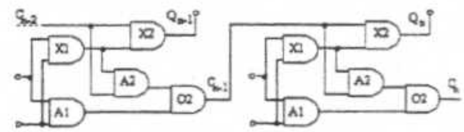

Figure 5. *n*-bit adder. X_1,X_2 denote *exclusive-or* gates A_1,A_2 are *and* gates, and O_1 is an *or* gate.

Removing the edge connected to the point Cn-2 lead to the system structural graph decomposition into disconnected subgraphs. Hence, measuring point *Cn-2* decomposes the system into two subsystems *Sub(b1,...,bn-2)* and *Sub(bn-1,bn)*. Suppose that *Cn-2=out(bn-2.O1)* is 0. So, the subsystem *Sub(b1,...,bn-2)* does not display the misbehavior. Hence, it may be excluded from the diagnosis. Consider the structural graph for *Sub(bn-1,bn)* depicted on Figure 6. Because one of the and-gate bn-1.A2 inputs is 0 the component bn-1.A2 behavior does not depend on the value assigned to the variable bn-1.x1. So, correspondent edge is removed from structural the graph *Sub(bn-1,bn)*.

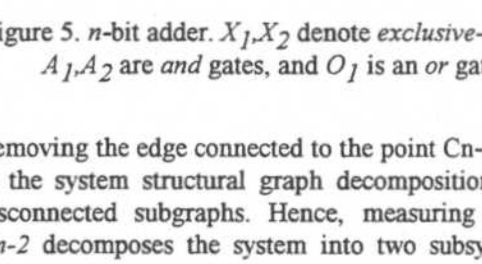

Figure 6. Structural graph for *Sub*(b_{n-1},b_n)

The best next measurement is *Cn-1* because it divides the set of suspected components into equal parts. Suppose *out(bn-1.O1)* is 0. In this case the component *bn.A2* behavior does not depend on the value assigned to its output connected to the variable *bn.X1*. So, resulting conflict graph is decomposed into three disconnected subgraphs *G1=(bn-1.A1, bn-1.A2, bn-1.O1)*, *G2=(bn.X1, bn.X2)*, *G2=(bn.A1, bn.A2, bn.O1)*. The subsystems *Sub(bn-1.A1, bn-1.A2, bn-1.O1)* and *Sub(bn.A1, bn.A2, bn.O1)* do not display the misbehavior. So, they are ruled out from the diagnosis. Only the subsystem *Sub(bn.X1,bn.X2)* needs to be decomposed. After measuring the component *bn.X1* output the diagnosis process stops.

6 Related works

Many work has been done to deal with the complexity of model-based diagnosis. Focused GDE approach [de Kleer, 1991] applies component failure probabilities and focuses on the probable diagnoses. Struss and Dressler [Struss and Dressler, 1989] use "physical negation" to rule out physically impossible diagnoses. Hierarchy approach [Genesereth, 1984], [Hamscher, 1990], [Mozetic, 1991] applies a multilevel hierarchical design that allows to restrict the number of components to be considered at each level.

An idea of diagnosis independence has been proposed in [Frietag and Friedrich, 1992]. They focus on independent diagnosis problems (IDP) and propose an algorithm for generating IDP. However, their concept of IDP ignores the symmetry of diagnosis independence, and their method does not apply the measurements for the IDP generating. A formal analisys shows that a system P-decomposition entails a system decomposition into IDP in meaning [Freitag and Friedrich, 1992]

7 Conclusions

The aim of this paper has been to reduce the diagnosis of a large system to the independent diagnosis of its subsystems. A formal definition for P-decomposition of a diagnosis problem into independent diagnosis subproblems has been proposed. However, we have not focused in developing purely theoretical results - instead we have demonstrated how these results may be applied in practical diagnosis. We have studied the measurements that enable *P*-decomposition. It has been shown that a strategy of measurement selection may be efficiently guided by "the first principle information" such as the system topology and current observations. Hence it may be an alternative to the traditional GDE approach. A formal algorithm for localization of multiple faults based on *P*-decomposition has been developed. In many cases the algorithm runs in logarithmic time compared to general diagnosis time.

References

[de Kleer and Williams, 1987], de Kleer, J. and Williams, B.C., Diagnosing multiple faults. Artificial Intelligence, 32(1), 1987, (pp. 97-130).

[de Kleer and Williams, 1989], de Kleer, J., Williams B.C., Diagnosis with behavioral modes. In Proc. of IJCAI-89, Detroit, 1989, (pp. 1324-1330).

[de Kleer, 1991], de Kleer, J., Focusing on probable diagnosis. In Proc. of 9th National Conf. on Artificial Intelligence, Anahaim, CA, 1991, (pp. 842-848).

[de Kleer et al., 1992], de Kleer J., Mackworth, A., Reiter, R., Characterizing diagnoses and systems. Artificial Intelligence, 56, 1992, (pp. 197-222).

[Friedrich et al., 1990] Friedrich G., Gottlob G., Nejdl W., Physical impossibility instead of fault models. In Proceedings AAAI 90, MIT Press.

[Freitag and Friedrich, 1992], Freitag H., Friedrich G., Focusing on independent diagnosis problems. In Proc. of Conf. on Knowledge Representation (KR'92), 1992.

[Genesereth, 1984], Genesereth, M., The use of design descriptions in automated diagnosis., Artificial Intelligence, 24 (1),1984, (pp. 411-436).

[Hamscher, 1990], Hamscher W., Diagnosing devices with hierarchic structure and known component failure modes. In Proceedings of 6th IEEE Conference on Artificial Intelligence Applications, Santa Barbara, CA, 1990, pp. 48-54

[Mozetic, 1991], Mozetic, I., Hierarchical model-based diagnosis. Int. J of Man Machine Studies, 35(3), 1991, (pp. 329-361).

[Reiter, 1987], Reiter, R., A theory of diagnosis from first principles. Artificial Intelligence, 32(1), 1987, (pp. 57-95).

[Struss and Dressler, 1989] P. Struss, O. Dressler, "Physical Negation"- Integrating Fault Models into the General Diagnostic Engine, Proc. IJCAI-9, Morgan Kaufmann.

[Tsybenko, 1994], Tsybenko, Y., Method of decomposition for diagnosing malfunctions in assumption-based model. In Proc. of the 2-th IEE Conf. Intelligent Systems Engineering, Hamburg, Germany, 1994, (pp. 286-291).

[Tsybenko, in prep], Tsybenko, Y., Model-based circuit decomposition. Submitted to IJCAI'95.