An Interval Model-Based Approach for Optimal Diagnosis Tree Generation

FAURE Pierre-Philippe [1,2] faure@laas.fr
TRAVE-MASSUYES Louise [1] louise@laas.fr
POULARD Hervé [1,2] poulard@laas.fr

[1] LAAS-CNRS
7, avenue du Colonel Roche
31077 Toulouse Cedex 4

[2] ACTIA
25, chemin de Pouvourville
31432 Toulouse

Abstract

In the automotive domain, the use of ECU (Electronic Control Unit) to control several functions (such as engine injection or ABS) increases. In order to diagnose such systems, diagnosis trees are built. These trees allow the garage mechanics to find the faulty component(s) by performing a set of tests (measurements) which has the lowest global cost as possible. Nowadays these diagnosis trees are hand made by human experts. This task requires more and more time and becomes more and more difficult as the complexity of electric circuits and mecatronic systems increases. Consequently, errors are not unusual in the resulting diagnosis trees. As a matter of fact, it becomes urgent to reduce the human intervention in the diagnosis tree generation process at the lowest.

Concerning the diagnosis problem, most of the electric circuits connected to the ECU can be viewed as a resistive net supplied by one voltage source. Moreover, one can emphasis that, on average, nineteen percent of failures are due to wires. This paper presents a method which computes an optimal diagnosis tree for this kind of electric circuits.

The first part of this paper deals with the model description, the generation of test and fault spaces corresponding to the studied system and the construction of a cross-table, using these models. A cross-table assigns a set of modalities for each couple (fault, test) coming from the above defined fault and test spaces. A modality is an interval of values taken by the considered test for the considered fault. A test is said to be binary if it has only two modalities, it is multi-modal otherwise. It is referred to as exclusive if, for each fault in the fault space, one unique modality is assigned to this test. Moreover, a weight is associated to each test to represent its cost, i.e. the difficulty for carrying it out.

Pattipati [Pattipati, 1990] has proposed a heuristic leading to an optimal diagnosis tree when used in combination with an adapted AO* algorithm for a cross-table corresponding to binary exclusive weighted tests. The second part of this paper addresses the issue of adapting this heuristic to multi-modality non-exclusive weighted tests which is the general case in our application domain. It is shown that the resulting heuristic also yields to an optimal diagnosis tree.

Key Words Model-Based Diagnosis, Diagnosis Tree Generation, AO* algorithm, Heuristic, Optimality

0. Introduction

In the automotive domain, the use of ECU (Electronic Control Unit) to control several functions (such as engine injection or ABS) increases. In order to diagnose such systems, diagnosis trees are built. These trees allow the garage mechanics to find the faulty component(s) by performing a set of tests (measurements) which has the lowest global cost as possible. Nowadays these diagnosis trees are hand made by human experts. This task requires more and more time and becomes more and more difficult as the complexity of electric circuits and mecatronic systems increases. Consequently, errors are not unusual in the resulting diagnosis trees. As a matter of fact, it becomes urgent to reduce the human intervention in the diagnosis tree generation process at the lowest.

Concerning the diagnosis problem, most of the electric circuits connected to the ECU can be viewed as a resistive net supplied by one voltage source. Moreover, one can emphasis that, on average, nineteen percent of failures are due to wires. This paper presents a method which computes an optimal diagnosis tree for this kind of electric circuits.

The first part of this paper deals with the model description, the generation of test and fault spaces corresponding to the studied system and the construction of a cross-table, using these models. A cross-table assigns a set of modalities for each couple (fault, test) coming from the above defined fault and test spaces. A modality is an interval of values taken by the considered test for the considered fault. A test is said to be binary if it has only two modalities, it is multi-modal otherwise. It is referred to as exclusive if, for each fault in the fault space, one unique modality is assigned to this test. Moreover, a weight is associated to each test to represent its cost, i.e. the difficulty for carrying it out.

Pattipati, in [Pattipati, 1990], has proposed a heuristic leading to an optimal diagnosis tree when used in combination with an adapted AO* algorithm for a cross-table corresponding to binary exclusive weighted tests. The second part of this paper addresses the issue of adapting this heuristic to multi-modality non-exclusive weighted tests which is the general case in our application domain. It is shown that the resulting heuristic also yields to an optimal diagnosis tree.

1. Problem Description and Case Study

Each fault of the fault space is defined by the set of initially failed components (without including those components whose fault results from a fault cascade effect). An occurrence probability is affected to each fault of the fault space. The test space is defined by all the accessible measurements. The accessibility of the measure provides a weight (also called cost of the test) which is affected to the test expressing the amount of work required to execute this test. By crossing over these two spaces, we obtain a table in which every element is the modality of the test for the corresponding fault. This modality is in the form of an interval value, providing all the possible measure values for the considered test and fault. In the general case, a test has several modalities, which discretise the quantity space of the test. From this table, called cross-table, we compute the optimal diagnosis tree. The cross-table is obtained along a model–based approach, in which the models are behavior interval models taking into account the tolerances of the components.

All the concepts and algorithms presented in this paper will be illustrated on a single circuit :

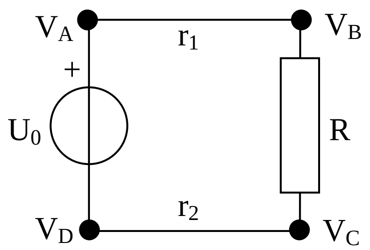

Figure 1.1 : Example

This circuit involves three different components :

- one ideal voltage source U_0
- one resistor R
- two wires characterized by their internal resistance r_1 and r_2

The wires must be considered as full components as most of the problem in the automobile electronics result from wire problems. This example is simple enough so as to allow us to develop the computations, illustrating however the approach without loss of generality. Despite its simplicity, it is highly representative of the application domain as it corresponds to the typical circuit of a sensor providing information to the ECU (Electronic Control Unit).

2. Partitionning the Test and Fault Spaces from the Models

2.1. Models

The modeling of the system is obtained along a standard component-connection approach. The component models describe the intrinsic behavior of every type of component and are stored in a library of component models, as in [Duffaut, 1994]. They are automatically assembled from the knowledge of the structure of the system to diagnose.

2.1.1. Component Models. The model of a component C is characterized by three sets:

the set of internal parameters, $\{P_1, \ldots, P_k\}$,

the set of (input and output) variables $\{V_1, \ldots, V_l\}$,

the set of constraints among the variables which define different behavior modes (BM), as in [Dague, 1987].

The same component may have several nominal and non-nominal behavior modes (nominal BM, non-nominal BM) defined by specific parameter and variable values. The different behavior modes are defined by mutually exclusive conditions on these values. A behavior mode is described by a set of equations binding parameters and variables. Moreover, according to component parameter value interval, component faulty or normal states are defined.

Example 2.1 : Generic resistor model R (with $[R_{min}, R_{max}]$ tolerance on the R value)

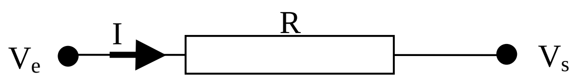

Figure 2.1 : Resistor representation

- Nominal behavior mode $I \le I_s$
  $V_e - V_s = R \times I$
  Component faulty state 1 : $R \in [0, R_{min}[$
  Component normal state 1 : $R \in [R_{min}, R_{max}]$
  Component faulty state 2 : $R \in]R_{max}, +\infty[$
- Non-nominal behavior mode $I > I_s$
  $I = 0$
  Component faulty state 1 : $R \approx +\infty$

2.1.2. System Model. As in [Darwiche, 1998], the system structural model represents the way the different components of the system are interconnected. The behavior model of the whole system is obtained by instanciating the component models and performing the adequate connections. In this model, all the required variables of the system are mentioned and all the components of the system are instanciated from the component models. The system model is organized so that the different nodes (i.e. connection point of degree greater than two) and branches (i.e. set of components between two consecutive nodes) can be identified. In particular, the Kirchhoff laws are automatically added to the model.

A system configuration (CONF) is defined as a specific BM for every component. As soon as one component is in a non-nominal BM the configuration is also non-nominal (non-nominal CONF).

Example 2.2 : One non-nominal configuration

Let us consider the fault scenario in which the voltage source has had a pulse which is sufficient to deteriorate the resistor ($I > I_s$). The system configuration is hence given by : wire r_1 in nominal behavior mode, wire r_2 in nominal behavior mode and resistor in non-nominal behavior mode since the constraint $I \le I_s$ is violated.

A behavioral equation obtained from the component models is associated to each component of the system. Moreover, knowing how the components are interconnected from the system model, the different physical equations linking variables and parameters, which just-determine the system, can be written.

If the physical equations are linear (it is the case in our example as in any resistive network since the Ohm and Kirchhoff laws are linear), the system model takes the following form :

$$P \times T = p$$

where P is a $n \times n$ square sparse matrix whose elements involve the system resistive parameters, n is the number of tests, T is the test vector and p is a vector whose elements involve the system electromotive parameters.

The execution of the Gauss-Jordan algorithm, detailed in [Ciarlet, 1993], allows us to obtain the formal expression of each test as a function of the system parameters. The Gauss-Jordan algorithm is a direct method to solve linear equation systems. The principle of this algorithm consists in diagonalizing the P matrix in n iterations. At each iteration a non null Gauss pivot is chosen. Let a_{ii} , A, and $B = [b_1..b_n]^T$ be the current chosen Gauss pivot, the current matrix and, the second member vector, respectively.

$$A = \begin{bmatrix} 1 & & 0 & a_{1i} & .. & a_{1n} \\ & . & & \vdots & & \vdots \\ 0 & & 1 & a_{(i-1)i} & .. & a_{(i-1)n} \\ 0 & .. & 0 & a_{ii} & .. & a_{in} \\ \vdots & & \vdots & \vdots & & \vdots \\ 0 & .. & 0 & a_{ni} & .. & a_{nn} \end{bmatrix}$$

$$A^{'} = \begin{bmatrix} 1 & & 0 & a^{'}_{1(i+1)} & .. & a^{'}_{1n} \\ & . & & \vdots & & \vdots \\ 0 & & 1 & a^{'}_{i(i+1)} & .. & a^{'}_{in} \\ 0 & .. & 0 & a^{'}_{(i+1)(i+1)} & .. & a^{'}_{(i+1)n} \\ \vdots & & \vdots & \vdots & & \vdots \\ 0 & .. & 0 & a^{'}_{n(i+1)} & .. & a^{'}_{nn} \end{bmatrix}$$

Assuming that $a_{ii} \neq 0$, by applying

$$a_{ij} = a_{ij} / a_{ii} \quad \forall j \neq i \text{ , } b_i = b_i / a_{ii}$$

$$a_{kj} = a_{kj} - \left(a_{ij} \times a_{ki}\right) \quad \forall k \neq i \quad \forall j \neq i$$

$$b_k = b_k - \left(b_i \times a_{ki}\right) \quad \forall k \neq i$$

and then $a_{ii} = 1$ and $a_{ki} = 0 \quad \forall k \neq i$, the current matrix becomes $A^{'}$ and the current second member vector $B^{'} = [b^{'}_1..b^{'}_n]^T$.

At end of the Gauss-Jordan algorithm, the current matrix is the $n \times n$ identity matrix. Hence, the test expressions are directly obtained in the second member vector.

System model in this configuration is given by the following equation system:

$$\begin{cases} I = 0 \\ V_A - V_B = r_1 \times I \\ V_C - V_D = r_2 \times I \\ V_A = U_0 \\ V_D = 0 \end{cases}$$

2.2. Computation of the Formal Expressions of the Tests in every Behavior Mode

A test may be any measurement which can be performed by the garage mechanics with a multi-meter : resistance value, intensity or voltage. As a matter of fact, using all the possible tests is highly redundant and would be computationally unpractical. We assume that the set of tests is restricted to a set which guarantees the diagnosability of every considered fault. The problem of determining this set is out of the scope of this paper (see [Van der Velde, 1984] and [Basseville, 1987]). All these tests are given by the value of formal expressions involving several parameters or variables. Since the variables can be formally expressed from the parameters, the formal expressions can be brought back to involve only parameters. In the following, a test is indifferently used to refer to the formal expression or its value.

Example 2.3 : The matrix representation and test expressions for the normal configuration are:

- Matrix representation

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ -1 & 0 & 1 & 0 & r_1 \\ 0 & 0 & -1 & 1 & R \\ 0 & 1 & 0 & -1 & r_2 \end{bmatrix} \times \begin{bmatrix} V_A \\ V_D \\ V_B \\ V_C \\ I \end{bmatrix} = \begin{bmatrix} U_0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

- Test Expressions

$$\begin{cases} V_A = U_0 \\ V_D = 0 \\ V_B = \dfrac{(r_2 + R) \times U_0}{r_1 + r_2 + R} \\ V_C = \dfrac{r_2 \times U_0}{r_1 + r_2 + R} \\ I = \dfrac{U_0}{r_1 + r_2 + R} \end{cases}$$

2.3.Fault Space

The aim of this subsection is to reduce the entire fault space to a subset of faults which have the highest probabilities to occur.

2.3.1.Fault Classification. Pure Single Faults (PSF) are defined as faults whose origin is one unique abnormal component. PSF include also cascaded abnormal component faults. Independent Multiple Faults (IMF) represent faults in which several components are abnormal without affecting common parts of the system. Overlapping Multiple Faults (OMF) represent faults in which several components are abnormal and affect some common parts of the system.

Obviously, IMF can be seen as a set of PSF. Consequently, if a diagnosis tree is able to detect PSF, it is also able to detect IMF by using this tree as many times as required. In our application domain, OMF have an occurrence probability lower than any PSF occurrence probability.

Moreover, in the case of multiple faults, we assume that it is more probable that the faults occur in independent sub-systems. Therefore, OMF have an occurrence probability lower than any IMF occurrence probability. This can summarized as follows :

$$\Pr(PSF) > \Pr(IMF) > \Pr(OMF)$$

This paper presents our approach to deal with the PSF and consequently with the IMF as well. In practice, the occurrence probability is obtained from the Mean Time Between Failures (MTBF) given by the designer, as shown in [Srinivas, 1994]. The PSF set is obtained by considering one by one all the possible faulty states for every component. The no fault situation (i.e. all the components behaving normally) is processed as a PSF, and it is referred to as a specific PSF. A PSF hence indifferently refers to the configuration of the system (in terms of the behavior mode of every component) or to the fault itself. The IMF case is processed as subsequent PSF.

3. Test and Fault Spaces Cross-Table Generation

The aim of this section is to obtain the cross-table between test and fault spaces. Every element of the table is the modality of the test for the corresponding PSF. This modality provides all the possible measure values for the considered test and PSF. In our case, a PSF is modeled by a component parameter taking its value in a non normal numeric interval. The normal interval is obtained from the normal value plus or minus a tolerance. For instance, a resistor R of normal value R_0 will give rise to three following cases :

$$\left(R \in [0, R_0 - \varepsilon]\right) \vee \left(R \in [R_0 - \varepsilon, R_0 + \varepsilon]\right)$$
$$\vee \left(R \in [R_0 + \varepsilon, +\infty]\right)$$

where ε represents the tolerance on the R_0 parameter numeric value.

Moreover, each PSF is defined by one component faulty state, so, one non-normal parameter interval. If, for the current CONF, and for the studied PSF, one limit of the CONF is reached for one of the system variable. So, it is necessary to divide this PSF into two PSF : the first one describes the originally studied PSF with the non-normal parameter interval such that this limit is never reached, and the second one described the originally studied PSF with the non-normal parameter interval such that this limit is always reached.

Consequently, this table is obtained by simulation, as for the tool proposed in [Seibold, 1994], according to the algorithm 3.1., called simulation algorithm.

The computation of the formal expression of the tests is performed within an interval optimization framework, using Classical Interval Analysis [Moore, 1979] and also Modal Interval Analysis [SIGLA/X, 1998]. For the particular formal expressions obtained for resistive networks supplied by one unique voltage source, it is possible to reach exact optimization values (the proof of this property is out of the scope of this paper).

From the algorithm 3.1., the value domain of every test is partitioned into interval values which may be associated to

PSF. The exact bounds of the interval values are approximated according to the precision of the measurement instrument. The resulting partition defines the modalities of the test, as shown in example 3.3..

Algorithm 3.1. : Simulation algorithm

```
Procedure Main_Treatment()
Begin
   For every Nominal-CONF do
      For every PSF do
         Recursive_Treatment (Nominal-CONF,PSF);
      End For
   End For
End

Procedure Recursive_Treatment (CONF,PSF)
Begin
   Matrix expression of the system;
   Formal expressions of the tests;
   Optimization of the formal expressions of the tests;
   If (at least one test value interval contains a CONF limit) then
      For each of the 2^n sets of n intervals obtained with n couples of
       intervals (under and over each of the n reached limits) do
         Evaluate the non-normal parameter interval;
         Create New-PSF with this new non-normal parameter interval;
         CONF:= change, in CONF, the non-consistent BM(s) into the
          consistent ones according to the studied set of n intervals;
         Recursive_Treatment (CONF, New-PSF);
      End For
   Else store PSF with CONF in cross-table;
   End If
End
```

Example 3.2. : Nominal configuration and over voltage PSF. The over voltage PSF is defined by $U_0 \in [2.2,+\infty]$. Let us assume that normal values for the other parameters are : $r_1 \in [0,2]$, $r_2 \in [0,2]$ and $R \in [2350,2670]$.

Moreover, the nominal CONF is constrained by the following condition : $I \le 2\times10^{-3}$. The optimization of the test I formal expression in the nominal CONF gives the interval $I \in [0.822\times10^{-3},+\infty]$.

So, since $2\times10^{-3} \in [0.822\times10^{-3},+\infty]$, we have to divide the PSF into PSF1 defined by $I \in [0.822\times10^{-3},2\times10^{-3}]$ and PSF2 defined by $I \in [2\times10^{-3},+\infty]$.

According to the optimization of the test I formal expression, we have :

$I \in [0.822\times10^{-3},2\times10^{-3}] \Rightarrow U_0 \in [2.2,5.348]$

$I \in [2\times10^{-3},+\infty] \Rightarrow U_0 \in [4.7,+\infty]$.

Hence, the two over voltage PSF obtained are defined as follows :

- PSF1 defined by $U_0 \in [2.2,5.348]$ in nominal CONF
- PSF2 defined by $U_0 \in [4.7,+\infty]$ in the corresponding non nominal CONF obtained from the nominal CONF by replacing the nominal resistor behavior mode (characterized by $I \le I_s$) by the non nominal one (characterized by $I > I_s$).

Example 3.3. : Cross-table

	Component Fault	**CONF**	**V_A**	**V_B**	**V_C**	**I**
F_0	$\varnothing$	Nominal	[1.8,2.2]	[1.798,2.2]	$[0,1.8\times10^{-3}]$	$[6.7\times10^{-4},9.36\times10^{-4}]$
F_1	$U_0\in[0,1.8]$	Nominal	[0,1.8]	[0,1.8]	$[0,1.5\times10^{-3}]$	$[0,7.6\times10^{-4}]$
F_{21}	$U_0\in[2.2,5.348]$	Nominal	[2.2,5.348]	[2.198,5.348]	$[0,4.5\times10^{-3}]$	$[8.22\times10^{-4},2\times10^{-3}]$
F_{22}	$U_0\in[4.7,+\infty]$	Non nominal	$[4.7,+\infty]$	$[4.7,+\infty]$	0	0
F_3	$r_1\in[2,+\infty]$	Nominal	[1.8,2.2]	[0,2.198]	$[0,1.869\times10^{-3}]$	$[0,9.35\times10^{-4}]$
F_4	$r_2\in[2,+\infty]$	Nominal	[1.8,2.2]	[1.798,2.2]	$[1.34\times10^{-3},2.2]$	$[0,9.35\times10^{-4}]$
F_{51}	$R\in[0,1100]$	Nominal	[1.8,2.2]	[1.796,2.2]	$[0,4.89\times10^{-3}]$	$[7.6\times10^{-4},2\times10^{-3}]$
F_{52}	$R\in[896,2350]$	Non nominal	[1.8,2.2]	[1.8,2.2]	0	0
F_6	$R\in[2670,+\infty]$	Nominal	[1.8,2.2]	[1.798,2.2]	$[0,1.646\times10^{-3}]$	$[0,8.22\times10^{-4}]$

For the test V_A, assuming that the voltage precision of the multi-meter is 1×10^{-1} V, the following modalities are obtained :

M1	**M2**	**M3**	**M4**	**M5**
$V_A\in[0,1.8]$	$V_A\in[1.8,2.2]$	$V_A\in[2.2,4.7]$	$V_A\in[4.7,5.3]$	$V_A\in[5.3,+\infty]$
F_1	F_0 F_3 F_4 F_{51} F_{52} F_6	F_{21}	F_{21} F_{22}	F_{22}

4. Optimal Diagnosis Tree Generation

First of all, this section defines accurately the objective function which is used to compare diagnosis trees. Its optimization provides the optimal diagnosis tree. This function combines the faults a priori occurrence probability with the costs of the tests to be performed to isolate the faults. The cost of a given test is given as a weight and represents the amount of work required in the garage to perform the test. These costs are assumed to be available for all the tests. Then, a method for double modality tests is presented and an extension of this method to non-exclusive multi-modality tests is suggested. A test is said to be exclusive if any pair of candidate fault sets corresponding to two distinct modalities for the concerned test do not share any fault. This method is based on the AO* algorithm with an adequate heuristic. The Huffman algorithm is at the heart of the heuristics presented for both cases.

The contribution of this paper is to propose a proof of the optimality of the Huffman algorithm for the diagnosis tree generation problem with binary exclusive unit weighted tests, what is not clearly done in [Pattipati, 1990]. Moreover, this proof is used to demonstrate the optimality of our multi-modality extension algorithm for the diagnosis tree generation problem with multi-modality exclusive unit weighted tests.

4.1. Optimality

Let us assume a diagnosis tree T which separates n PSF. Let us call $\{F_1,\ldots,F_n\}$ the n PSF, P_i the prior occurrence probability of each F_i fault with $\sum_{i=1}^{n} P_i = 1$, $\{L_1(F_i),\ldots,L_{m_i}(F_i)\}$ the m_i leaves of the diagnosis tree T which contain the F_i fault, P_j^i the probability of F_i on the j^{th} leave with $\sum_{j=1}^{m_i} P_j^i = 1$, and $C\big(L_j(F_i)\big)$ the sum of the costs of the tests required to connect the j^{th} leave containing the F_i fault from the root of the diagnosis tree T.

Then, the objective function of a diagnosis tree T to optimize, noticed $K(T)$, can be computed in two steps. The first step consists in evaluating $C(F_i)$ which is a weighted sum, on the m_i leaves containing the F_i fault, of $C\big(L_j(F_i)\big)$. The second step is the computation of the $K(T)$ value as a weighted sum, on the n faults, of $C(F_i)$.

$$\begin{cases} K(T) = \sum_{i=1}^{n} P_i \times C(F_i) \\ C(F_i) = \sum_{j=1}^{m_i} P_j^i \times C\big(L_j(F_i)\big) \end{cases}$$

4.2. AO* Algorithm

4.2.1. Principle. An AO* algorithm is based on an AND/OR tree T. An AND/OR tree contains AND nodes which have the property to be true if all their children are true and OR nodes which have the property to be true if at least one of their children is true. Because of the complexity, T is not generally explicitly represented so, it is said to be implicit. Only the ground elements, such as the generation of their whole combinatory gives explicitly T, are available. So, the goal of this algorithm is, starting form this ground element knowledge about T, find the optimal sub-tree T^* of T which connects a sub-set of leaves of T to its root.

Let us come back to the diagnosis tree problem. An OR node corresponds to a node labeled by a set of possible PSF and an AND node represents a test, each of its branches being the possible modalities of the test, corresponding indeed to a set of PSF. Hence, a diagnosis tree can be seen as an AND/OR tree in which each OR node has one AND node child and each AND node has n OR node children, n being the modality cardinality (i.e. number of modalities) of the test. The root of a diagnosis tree is an OR node whose label contains all the PSF to discriminate. T is the implicit tree that could be built by considering all the AND nodes as children of each OR node instead of only one. During the algorithm, T' is an explicit current sub-tree of T, which corresponds, at end of the algorithm, to the optimal sub-tree T^* of T. In other words, T^* provides the optimal sequence of tests to be performed in order to discriminate all the PSF.

4.2.2.Heuristic. The AO* algorithm uses a heuristic to orient the search. A heuristic is a function which gives an estimate, for each node, of the cost of the subsequent sub-tree, considering this node as its root. Let us call a the studied node, $h^*(a)$ the exact evaluation of the corresponding sub-tree and $h(a)$ the one estimated by the heuristic. It has been proven that if for each node a $h(a) < h^*(a)$ (admissibility property of h, see [Bagchi, 1983]) then the algorithm converges to the optimal sub-tree T^* of T. Moreover, the convergence rate is directly related to the quality of the heuristic h. The closer $h(a)$ of $h^*(a)$, the lowest is the number of useless nodes (i.e. nodes which do not appear in the optimal sub-tree T^*) expanded during the algorithm. Let us analyze the two extreme cases :

- if there is no heuristic then the implicit tree T must be completely explored to find the optimal sub-tree T^*,
- with a heuristic h such as $h(a) = h^*(a)$, then the optimal sub-tree T^* is obtained immediately without expanding any useless node.

It is important to outline that, in the diagnosis tree problem, since each OR node must have only one AND node child, only the OR nodes need to be estimated. In consequence, the heuristic concerns the OR nodes only.

4.2.3. Huffman Algorithm. For a set F of n faults F_i with their respective a priori occurrence probabilities P_i and a set S of m binary exclusive tests S_j affected with the same weight $c_j = 1$, the Huffman algorithm builds an optimal binary diagnosis tree T^*.

Since the tests are exclusive, there is only one leave $L(F_i)$ for each fault F_i and the objective function $K(T)$ becomes the weighted sum of the depth of T.

$$K(T) = \sum_{i=1}^{n} P_i \times C(F_i)$$

where, given that the tests are unit weighted, $C(F_i) = C(L(F_i))$ corresponds to the level in T of the leave which contains F_i.

Property 4.1.

A binary diagnosis tree T, admissible solution of the considered problem, verifies the node ordering condition , if : each node of the i^{th} level has an *a priori* occurrence probability greater than the one of any node appearing on a level j such that $d \geq j > i \geq 0$, where d represents the diagnosis tree depth.

Then, such binary diagnosis tree T is optimal according to the objective function $K(T)$.

The proof of property 4.1. can be found in appendix A.

Algorithm 4.2. : Huffman algorithm

```
Begin
       i:=n;
       Create n leaves corresponding to the n faults;
       While (i>1) do
            Order the i faults by their increasing occurrence probability;
            Pi-1:=Pi-1+Pi; (Pi and Pi-1 are the two least probabilities)
            Create new node Fi-1 father of Fi and old Fi-1 nodes;
            Suppress Fi fault and Pi probability from the fault set;
            i:=i-1;
       End While
       return node F1 root of the created optimal tree;
End
```

Theorem 4.3.
In case of exclusive and unit weighted binary tests, the Huffman algorithm creates an optimal complete binary diagnosis tree according to the objective function $K(T)$.

The proof of theorem 4.3. can be found in appendix B.

4.3.Heuristic in Case of Exclusive Binary Tests with Different Costs

As seen previously, the Huffman algorithm builds an optimal diagnosis tree for exclusive binary tests S_j whose weights are $c_j = 1$. What about the optimality of the obtained diagnosis tree when the different possible tests have different weights? In this case, the diagnosis tree is not optimal anymore. So, the idea is to use an AO* algorithm with an admissible heuristic taking into account the different test weights in order to reach optimality.

Let a be any OR node and K^* the optimal objective function value of the optimal diagnosis tree T^* obtained by the Huffman algorithm for the set of faults which are contained in a assuming all exclusive unit weighted ($c_j = 1$) binary test S_j. The m available tests are ordered by increasing weights such as $0 < c_1 <...< c_m$.

The admissible heuristic h_P proposed by Pattipati is then expressed as follows :

$$h_P(a) = \sum_{j=1}^{K'} c_j + [K^* - K'].c_{K'+1}$$ where K' is the integer part of K^*.

This heuristic is equivalent to build a Huffman tree with the tests of lowest weight. Consequently, this heuristic verifies the admissibility property, and hence, leads to the optimal tree by using the AO* algorithm.

4.4.Extension to Non Exclusive Multi-Modality Tests

4.4.1. Non Exclusivity Assumption. It is obvious that the non exclusive test assumption makes the discrimination of a given fault set with the same test set more difficult. That is, the optimal diagnosis tree obtained by the Huffman algorithm with the non exclusive test assumption is deeper than the one with the exclusive test assumption. Consequently, assuming a priori the test exclusivity assumption in an admissible heuristic h does not affect its admissibility property. However, it is important to outline that, when some of the tests are actually non-exclusive, this decreases the quality of the heuristic (i.e. the distance between h and h^* increases).

4.4.2. Multi-Modality Assumption. At the contrary, the test multi-modality assumption makes the discrimination of a same fault set with a same test set easier than the binary test assumption. That is, optimal diagnosis tree obtained by Huffman algorithm with the test binary assumption is deeper than the one with the multi-modality test assumption. Consequently, considering in priority tests which have the biggest modality cardinality in an admissible heuristic h, reduces the depth of the diagnosis tree and, so, allows to keep its admissibility property.

Let T^* be the optimal diagnosis tree obtained by Huffman algorithm for the considered fault set assuming exclusive unit weighted ($c_j = 1$) binary tests S_j. The multi-modality extension algorithm modifies T^* so as to obtain T_M^* the optimal diagnosis tree for the considered fault set assuming exclusive unit weighted ($c_j = 1$) tests S_j having the biggest available modality cardinalities.

Algorithm 4.4. : Multi-modality extension algorithm

```
Begin
       Order the m test modality cardinalities such as (M1≥...≥Mm≥2);
       i:=1;
       Finish:=false;
       While ((Finish=false)AND(Mi>2)) do
          Finish:=true;
          For all non-leave nodes of level (i-1) do
                 Card:=2;
          While ((card<Mi)AND
       (it exists at least one non leave child)) do
                        break the biggest child;
                        card:=card+1;
                        reorder the children of the studied node;
                 End While
                 If (card=Mi) then Finish:=false;
       End For
       i:=i+1;
       End While
End
```

Theorem 4.5.
In case of exclusive unit weighted multi-modality tests, from the optimal binary diagnosis tree obtained by Huffman algorithm T^*, the multi-modality extension algorithm creates an optimal diagnosis tree T_M^* according to the objective function $K(T)$.

The proof of theorem 4.5. can be found in appendix C.

4.4.3. Heuristic in Case of Non Exclusive Multi-Modality Tests. As seen previously, to take into account different test weights, an AO* algorithm is used in order to lead to an optimal diagnosis tree.

Contrarily to the case of exclusive multi-modality tests, Pattipati heuristic is not applied directly from the optimal binary diagnosis tree T^* obtained by the Huffman algorithm, but from the optimal multi-modality diagnosis tree T_M^* obtained by the multi-modality extension algorithm applied on the optimal binary diagnosis tree T^* obtained by the Huffman algorithm initially computed. The m available tests are ordered by their increasing weights such as $0 < c_1 < ... < c_m$.

The admissible heuristic h_P proposed by Pattipati is then expressed as follows :

$$h_P(a) = \sum_{j=1}^{K'} c_j + [K^* - K'].c_{K'+1}$$ where K' is the integer part of K^*.

This heuristic is equivalent to build a Huffman tree with the highest available modality cardinality tests affected by the lowest available test weights. Consequently, this heuristic reaches the admissibility property, and so, leads to the optimal tree by using the AO* algorithm.

5. Conclusion

The first part of the paper applies the standard model-based component-connection approach to the representation of systems in the linear electronic circuit domain. The notions of nominal and non-nominal behavior mode for a component and the corresponding notions of nominal and non-nominal configuration at the level of the whole system, as well as the notions of normal and faulty states are clearly stated and distinguished. Moreover, an interesting fault classification in PSF, IMF and OMF is defined. According to these definitions, an adapted simulation algorithm allows to obtain the cross-table.

The second part of the paper discusses the adaptation of the method proposed by [Pattipati, 1990] to the specific resistive net problem. This method is based on an AO* algorithm with an admissible heuristic computed from the Huffman algorithm which converges to the optimal tree faster than the entropy based heuristic for exclusive binary tests affected by different weights. The cross-table may have non-exclusive multi-modality tests affected by different weights. It is shown how to modify the computation of the Pattipati heuristic in order to have an admissible heuristic allowing to obtain, by using an AO* algorithm, an optimal diagnosis tree for non-exclusive multi-modality tests affected by different weights.

Unfortunately, the more an admissible heuristic is able to take into account several constraints (as it is the case here), the more the quality of this heuristic decreases. Obviously, this admissible heuristic converges more slowly to the optimal diagnosis tree with non-exclusive multi-modality tests, than the Pattipati heuristic with exclusive binary tests. However, as the diagnosis trees are built off-line, the convergence speed to the optimal diagnosis tree is not a strong requisite in our problem.

Appendix A

Property 4.1.

A binary diagnosis tree T, admissible solution of the considered problem, verifies the node ordering condition , if : each node of the i^{th} level has an *a priori* occurrence probability greater than the one of any node appearing on a level j such that $d \geq j > i \geq 0$, where d represents the diagnosis tree depth.

Then, such binary diagnosis tree T is optimal according to the objective function $K(T)$.

Proof

As seen previously, the optimality, under test exclusivity and unit weighted assumptions, can be expressed as the minimization of the following objective function :

$$K(T) = \sum_{i=1}^{n} P_i \times C(F_i)$$

For any diagnosis tree T which does not verify the node ordering condition, it exists at least one sequence of permutations (which establishes the node ordering condition between two specific nodes) to apply on T in order to obtain the optimal diagnosis tree T^* which, for all of its nodes, verifies the node ordering condition.

Each application of one of these permutation reduces the objective function value of the studied diagnosis tree. Actually, by considering two faults F_1 and F_2 with occurrence probabilities P_1 and P_2 such as $P_1 > P_2$. Let i and j be two different levels of the studied diagnosis tree T_1 such as $d \geq j > i \geq 0$, where d represents the diagnosis tree depth.

For this diagnosis tree T_1, F_2 is placed on level i and F_1 on level j, the objective function $K(T_1)$ gives :

$$K(T_1) = (j \times P_1) + (i \times P_2) + \sum_{k=3}^{n} P_k \times C(F_k) =$$

$$(i \times (P_1 + P_2)) + ((j-i) \times P_1) + \sum_{k=3}^{n} P_k \times C(F_k)$$

The diagnosis tree T_2 is obtained by performing the permutation which consists in placing F_1 on level i and F_2 on level j, the objective function $K(T_2)$ gives :

$$K(T_2) = (j \times P_2) + (i \times P_1) + \sum_{k=3}^{n} P_k \times C(F_k) =$$

$$(i \times (P_1 + P_2)) + ((j-i) \times P_2) + \sum_{k=3}^{n} P_k \times C(F_k)$$

As $P_1 > P_2$, $K(T_1) \geq K(T_2)$.

So, the application of one permutation according to the node ordering condition reduces the objective function value.

Moreover, once all the nodes of the current tree are sorted by level, this tree verifies the node ordering condition and no permutation, according to the node ordering condition, is applicable. Consequently, the objective function value of this diagnosis tree can not be reduced and is optimal. Hence, this tree is the optimal diagnosis tree T^*. ■

Appendix B

Theorem 4.3.

In case of exclusive and unit weighted binary tests, the Huffman algorithm creates an optimal complete binary diagnosis tree according to the objective function $K(T)$.

Proof

From the property 4.1., it can be deduced that the two least occurrence probability faults of the considered fault set are always placed on the deepest level of the optimal diagnosis tree corresponding to this fault set.

Actually, at each iteration of the Huffman algorithm, the two least occurrence probability faults are found, these two faults are suppressed from the studied fault set and a new fault is created which has, as occurrence probability, the sum of these two least occurrence probabilities. This new virtual fault is, in the complete binary tree, the father node of these two precedent least occurrence probability faults. So, at each iteration of the Huffman algorithm, a new fault set is considered and the two least occurrence probability faults of this set are affected to the two deepest leaves of the optimal diagnosis tree corresponding to this set.

Consequently, by reapplying recursively the previous reasoning and treatment, node ordering condition is always verified and, by the previous property, optimality is ensured.

Moreover, an optimal binary diagnosis tree T^* obtained by Huffman algorithm, whose objective function value is K^* the optimal value for the considered problem, is always complete (i.e. T^* does not contain empty leaves) by construction. ■

Appendix C

Theorem 4.5.
In case of exclusive unit weighted multi-modality tests, from the optimal binary diagnosis tree obtained by Huffman algorithm T^*, the multi-modality extension algorithm creates an optimal diagnosis tree T_M^* according to the objective function $K(T)$.

Proof
As seen previously, the optimality under test exclusivity assumptions can be expressed as the minimization of the following objective function :

$$K(T) = \sum_{i=1}^{n} P_i \times C(F_i)$$

First, it can be proven that, at each step of the multi-modality algorithm, the current tree T is optimal according to the objective function $K(T)$. To reach this optimality, the current tree T must verify the node ordering condition (according to the Property 4.1.).

Let A be the studied node during the multi-modality algorithm and T_0 the three first levels of the binary sub-tree whose A is the root. On the figure 4.1., the general case is considered, where these three levels are assumed to be full (the maximum node cardinality that can be placed on a binary tree of three levels is 7).

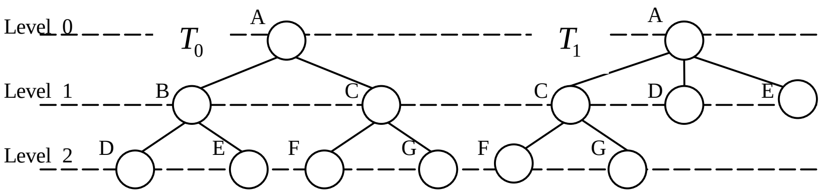

Figure 4.1. : Central treatment of the multi-modality extension algorithm

The relations between the nodes $\{A, B, C, D, E, F, G\}$, involved by the construction of the tree T_0 according to the Huffman algorithm, are expressed below and verify the node ordering condition (according to property 4.1. and theorem 4.3.).

$$\begin{cases} \Pr(A) = \Pr(B) + \Pr a(C) \\ \Pr(B) = \Pr(D) + \Pr(E) \\ \Pr(C) = \Pr(F) + \Pr(G) \\ \Pr(B) \geq \Pr(C) \\ \Pr(D) \geq \Pr(E) \\ \Pr(F) \geq \Pr(G) \\ \Pr(C) \geq \Pr(D) \\ \Pr(E) \geq \Pr(F) \end{cases}$$

The tree T_1 is obtained by applying the central treatment (break the biggest child; and reorder the children of the studied node;) on node A of the tree T_0. The relations between the nodes of the resulting tree T_1 are expressed below and verify also the node ordering condition.

$$\begin{cases} \Pr(A) = \Pr(C) + \Pr(D) + \Pr(E) \\ \Pr(C) = \Pr(F) + \Pr(G) \\ \Pr(C) \geq \Pr(D) \geq \Pr(E) \\ \Pr(F) \geq \Pr(G) \\ \Pr(E) \geq \Pr(F) \end{cases}$$

Consequently, the multi-modality extension algorithm applied on a binary diagnosis tree obtained by Huffman algorithm verifies this node ordering condition at each of its steps and, so, ensures optimality of the objective function $K(T)$.

Secondly, it is important to prove that affecting the i^{th} highest modality cardinality test to the i^{th} level of the tree is the optimal way to decrease the depth of the tree and so to decrease the objective function $K(T)$.

Let S_1 and S_2 be two tests of modality cardinality m_1 and m_2 with $m_1 > m_2$ and a set of n faults $\{F_1, \ldots, F_n\}$ affected with the occurrence probabilities

P_k such that $\sum_{k=1}^{n} P_k$, $P_1 > ... > P_n$ and $m_1 < n \le (m_1 \times m_2)$.

In the diagnosis tree T_1, S_1 is applied just before S_2, the number of faults n is such that $n = m_1 - i + (i \times m_2)$ with $1 < i \le m_1$ and $K(T_1) = \sum_{k=1}^{m_1 - i} P_k + \left(2 \times \sum_{k=m_1-i+1}^{n} P_k\right)$.

In the diagnosis tree T_2, S_2 is applied just before S_1, the number of faults n is such that $n = m_2 - j + (j \times m_1)$ with $1 < j \le m_2$ and $K(T_2) = \sum_{k=1}^{m_2 - j} P_k + \left(2 \times \sum_{k=m_2-j+1}^{n} P_k\right)$.

So, if $m_1 - i \ge m_2 - j$ then $K(T_1) \le K(T_2)$ else $K(T_1) > K(T_2)$

$$m_1 - i \ge m_2 - j \Leftrightarrow m_1 - \frac{n - m_1}{m_2 - 1} \ge m_2 - \frac{n - m_2}{m_1 - 1}$$

$$\Leftrightarrow m_1(m_2 - 1)(m_1 - 1) - (n - m_1)(m_1 - 1)$$

$$\ge m_2(m_2 - 1)(m_1 - 1) - (n - m_2)(m_2 - 1)$$

$$\Leftrightarrow m_1^2 m_2 - m_1^2 - m_1 m_2 + m_1 - n m_1 + n + m_1^2 - m_1$$

$$\ge m_2^2 m_1 - m_2^2 - m_1 m_2 + m_2 - n m_2 + n + m_2^2 - m_2$$

$$\Leftrightarrow m_1(m_1 m_2 - n) \ge m_2(m_1 m_2 - n)$$

Since $n \le m_1 m_2$,

$\Leftrightarrow m_1 \ge m_2$ which is true by hypothesis.

Consequently, $K(T_1) \le K(T_2)$, hence, it is optimal to apply the i^{th} highest modality cardinality test to the i^{th} level of the tree. ■

References

[Moore, 1979] MOORE R. E. "Methods and Applications of Numerical Analysis", SIAM Studied In Applied Mathematics, Philadelphia : An up to date text on Classical Interval Analysis.

[Bagchi et al., 1983] A. BAGCHI and A. MAHANTI "Admissible Heuristic Search in AND/OR Graphs" Theoretical Computer Science, No. 24, pages 207-219, 1983.

[Van der Velde et al, 1984] Van der Velde W.E., Carignan C.R., Number and placement of control system components considering possible failures, Journal of Guidance, Control, Dynamics, 7(6), p. 703-709, 1984.

[Basseville et al, 1987] Basseville M., Benveniste A., Moustadikes G., Rougée A., "Optimal sensor location for detecting changes in dynamical behavior", IEEE Transactions on Automatic Control, AC 32(12), p. 1067-1075, December 1987.

[Dague et al., 1987] P. DAGUE, O. RAIMAN, P. DEVES "Troubleshooting : when modeling is the trouble" In proceedings AAAI Seattle 1987, pages 600-605.

[Pattipati et al., 1990] K.R. PATTIPATI and M.G. ALEXANDRIDIS "Application of Heuristic Search and Information Theory to Sequential Fault Diagnosis" IEEE Transaction on Systems Man and Cybernetics, Vol.20, No.4 July/August 1990.

[Ciarlet, 1993] P.G. CIARLET "Introduction à l'analyse numérique matricielle et à l'optimisation" Edition MASSON, 1993.

[Lee et al., 1993] M.H. LEE, A.R.T. ORMSBY "Qualitatively modeling the effects of electrical circuit faults" Artificial Intelligence in Engineering, volume 8, No 4, 1993, pages 293-300.

[Duffaut, 1994] O. DUFFAUT " Problématique Multi-Modèle pour la génération d'Arbres de Tests : Application au Domaine de l'Automobile" Rapport de Thèse Automatique ENSAE, 1994.

[Seibold, 1994] W. SEIBOLD "A Model-Based Diagnosis and Simulation System in Practical Use : The Concept of ro•on" Fifth International Workshop on Principles of Diagnosis 1994 (DX94).

[Srinivas, 1994] S. SRINIVAS "On the computation of *a priori* failure probabilities from empirical measures of reliability" Fifth International Workshop on Principles of Diagnosis 1994 (DX94).

[Darwiche, 1998] A. DARWICHE "Utilizing Device Non–Structural Properties in Structure-Based Diagnosis" Ninth International Workshop on Principles of Diagnosis 1998 (DX98).

[SIGLA/X, 1998] SIGLA/X Group "Modal Intervals : draft for a basic tutorial" research report IMA Informática y Matematica Aplicada department, University of Girona, October 1998.