AGENDA : Automatic GENeration of DiAgnosis trees

Pierre-Philippe FAURE[1,2], Xavier OLIVE[1,2], Louise TRAVE-MASSUYES[1] and Hervé POULARD[2]
[1] LAAS-CNRS, [2] ACTIA
respectively faure@laas.fr, xolive@laas.fr, louise@laas.fr and poulard@actia.fr

***Abstract*— In the automotive field, 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. The difficulty of doing this task increases due to the complexity of electronic circuits and mecatronic systems. Consequently, errors are not unusual and it becomes urgent to reduce the human intervention in the diagnosis tree generation process at the lowest. This paper presents the software application AGENDA which computes an optimal diagnosis tree for electronic circuits. In order to find a solution to the test sequencing problem, the used diagnosis method is based on a prediction algorithm and on the execution of an AO* algorithm which allows us to obtain an optimal diagnosis tree.**

I. INTRODUCTION

In the automotive field, the use of electronic systems to control several functions has been widely developed during these last years. These functions cover more and more different automotive areas such as engine control (fuel injection or ignition), road driving (ABS or suspensions), security (air-bags or seat-belt), or comfort (air conditioning or heating system).

These electronic systems are roughly composed of voltage supply sensors (potentiometer or temperature sensor) and actuators (electro-valve) linked to Electronic Control Units (ECU) by a wire harness.

The main task of the ECU is to elaborate and to send control signals to its actuators taking into account the signals received by its sensors. Moreover, the ECUs are equipped with an auto-diagnosis function which reliably detects which of the electronic circuits that are connected to this ECU are failing. However, the ECU is not able to localize precisely the faulty components even if it is able to detect the failed electronic circuit.

In order to diagnose such electronic circuits, diagnosis trees are built. These trees allow the garage mechanic to find the faulty component(s) by performing a sequence 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 electronic 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.

In order to automatically build these diagnosis trees from the design data supplied by the car manufacturer the software diagnosis application AGENDA (Automatic GENeration of DiAgnosis trees) has been developed.

AGENDA can be viewed as the non-interactive step (i.e. the failed system is not yet on the workshop test bench, however, knowledge about the possible faulty behaviors of this system is available) of an off-line diagnosis application (see [1]). The information required for the interactive step (i.e. the diagnosis session when the failed system is on the workshop test bench) is precompiled in the diagnosis tree delivered by AGENDA.

The aim of this paper is to briefly present the diagnosis method AGENDA, the open issues and the future work.

The first section describes the prediction process that provides the set of faults to discriminate, the set of tests that can be performed on the system and the corresponding "cross-table" from the design data supplied by the car manufacturer. Each cell of the "cross-table" corresponds to one of the possible couples (fault/test) and contains the possible outcomes of the test in occurrence of the fault.

The second section characterizes the specific test sequencing problem that has to be solved in order to obtain an optimal diagnosis tree from the created fault set, test set and corresponding "cross-table" (see [2]).

The third section discusses the performances of the AGENDA application and several interesting directions for future investigation are outlined in the fourth section.

II. PREDICTION PROCESS

The first step is to build a behavioral model of the system to be diagnosed is created from the design data supplied by the car manufacturer according to a classical component-oriented approach (see [3]).

Then, the set of faults that are considered for discrimination, and the set of tests that are considered available on the system are anticipated from this behavioral model.

At last, the possible outcomes of a given test when the system is in a given faulty state are computed according to a prediction algorithm that uses symbolic computation followed by multi-variable optimization on intervals.

A. Behavioral model

A behavioral model is characterized by a set Z of n_Z mode variables $\{z_1, ..., z_{n_Z}\}$, a set X of n_X state variables $\{x_1, ..., x_{n_X}\}$ and a set Y of n_Y parameters $\{y_1, ..., y_{n_Y}\}$.

n_M different modes Z_k with $k \in \{1, ..., n_M\}$ are defined as vectors of n_Z values assigned to each of the mode variables z_i with $i \in \{1, ..., n_Z\}$. In the same way, n_L parameter initializations Y_k with $k \in \{1, ..., n_L\}$ are defined as vectors of n_Y values assigned to each of the parameters y_i with $i \in \{1, ..., n_Y\}$.

For any $k \in \{1, ..., n_M\}$, Z_k is associated with one couple σ_k defined by one behavior $b_k(X, Y)$ and one parameter assignment vector Y_k as shown in equation 1. The behavior $b_k(X, Y)$ is expressed as a system of equations involving state variables of the set X, parameters of the set Y and mode variables of the set Z.

$$Z_k \Rightarrow \sigma_k = (b_k(X, Y), Y_k) \tag{1}$$

Consequently, a behavioral model is composed of the identification of the state variable set X, the parameter set Y and the mode variable set Z and the explicit association between any possible mode defined by Z_k and its corresponding couple $(b_k(X, Y), Y_k)$ with $k \in \{1, ..., n_M\}$.

Moreover, the set Z of mode variables may be divided into two subsets Z_E and Z_F such that $Z_E \cup Z_F = Z$ and $Z_E \cap Z_F = \emptyset$. Z_E represents the set of configuration mode variables which denote discrete state of an elementary entity (ON / OFF for a switch, for instance). Z_F denotes the set of faulty mode variables which denote the possible faulty modes, including the fault-free one, of an elementary entity.

A component mode Z_k is said to be faulty if at least one of the mode variables in Z_F is assigned to a faulty mode ; it is fault-free otherwise.

For any set of interconnected components, a structural model expresses the connections between these components as a system of equalities which equals two distinct state variables belonging to two distinct components (see [4]).

The structural model defines distinct sets of identical state variables belonging to different components. Each of these distinct component state variable sets defines a different system state variable.

Let Ψ be the system to be diagnosed defined as a set of n_Ψ individual components ψ_i with $i \in \{1, ..., n_\Psi\}$. The behavioral model of the system Ψ, called BM_Ψ, is built according to a component oriented approach. As shown in equation 2, this model is composed of both the behavioral models BM_{ψ_i} corresponding to the components ψ_i with $i \in \{1, ..., n_\Psi\}$ and the structural model of the system Ψ, called SM_Ψ, which describes the way these components are interconnected.

$$BM_\Psi = SM_\Psi \cup BM_{\psi_1} \cup ... \cup BM_{\psi_{n_\Psi}} \tag{2}$$

B. Fault and test sets anticipation

Let Ψ be the system to be diagnosed defined as the set of its n_Ψ individual components ψ_i with $i \in \{1, ..., n_\Psi\}$.

This subsection describes how the set F of the faults, the set R of the corresponding repairs and the set S of the tests that can be performed on the system are obtained from the behavioral model.

1) Fault set: For each component ψ_i, let Φ^i be the set of the n^i_Φ possible faulty modes. Let also $\Phi^i_{\neg AB}$ and Φ^i_{AB} be the set of the $n^i_{\neg AB}$ fault-free modes and the set of the n^i_{AB} faulty modes, respectively, such that $\Phi^i_{\neg AB} \cup \Phi^i_{AB} = \Phi^i$ and $\Phi^i_{\neg AB} \cap \Phi^i_{AB} = \emptyset$.

A faulty mode of the system Ψ, also called system fault, is defined as a n_Ψ dimension vector which associates to each component ψ_i one of its n^i_Φ possible faulty modes. Consequently, the set F of faults which may occur in the system Ψ is composed of $n_F = \prod_{i=1}^{n_\Psi} n^i_\Phi$ elements, called f_k with $k \in \{1, ..., n_F\}$.

2) Test set: Let X be the set of the n_X system variable state defined by the structural model of the system Ψ.

For each component ψ_i, let U^i be the set of the n^i_U possible configuration modes.

A configuration of the system Ψ is defined as a n_Ψ dimension vector which associates to each component ψ_i one of its n^i_U possible configuration modes. Consequently, the set E of possible configurations of the system Ψ is composed of $n_E = \prod_{i=1}^{n_\Psi} n^i_U$ elements.

A test is defined as a pair composed of a measurement description based on a subset of the system state variable set X and a subset of the possible system configuration set E on which all these configurations are equivalent (i.e. give the same outcome).

C. Prediction algorithm

This work assumes that the system to be diagnosed is an electronic circuit in the form of a resistance net supplied by one voltage source. For this system, let F be the set of the n_F considered faults and S the set of the n_S considered tests.

For any test in S and any fault in F, the aim of the prediction process is to provide the symbolic expression of the outcome of the test in the occurrence of the fault.

1) Symbolic matrix expression of the system model: As shown in figure 1, the symbolic matrix expression of the system model is in the form $A \times X = B$ where the square matrix A is decomposed into 9 blocks, called $A_{i,j}$ with $i \in \{1, ..., 3\}$ and $j \in \{1, ..., 3\}$.

The blocks $A_{1,2}$, $A_{1,3}$, $A_{3,1}$ and $A_{3,2}$ are null and $A_{1,1}$ is an identity (2×2) matrix.

The vector X is decomposed into 3 sub-vectors, called X_i with $i \in \{1, ..., 3\}$. X_1 is a 2 components vector such that the first component corresponds to the

$$\left(\begin{array}{c|c|c} A_{1,1} & A_{1,2} & A_{1,3} \\ \hline A_{2,1} & A_{2,2} & A_{2,3} \\ \hline A_{3,1} & A_{3,2} & A_{3,3} \end{array}\right) \times \left(\begin{array}{c} X_1 \\ \hline X_2 \\ \hline X_3 \end{array}\right) = \left(\begin{array}{c} B_1 \\ \hline B_2 \\ \hline B_3 \end{array}\right)$$

Fig. 1. Symbolic matrix expression model

ground of the system and the second one to the supply pin. X_2 corresponds to the other system potential points. X_3 corresponds to the different intensities of the system.

The vector B is decomposed into 3 sub-vectors, called B_i with $i \in \{1, ..., 3\}$. B_1 is a 2 components vector such that the first component is 0 and the second one U_{Supply} (i.e. electromotive power value of the system voltage source). B_2 and B_3 are null sub-vectors.

According to this decomposition of the symbolic matrix expression, the Kirchhoff's law equations are stated in the last lines of the matrix expression ($A_{3,3}$ in A, X_3 in X and B_3 in B). In the same way, the Ohm's law equations ($A_{2,1}$, $A_{2,2}$ and $A_{2,3}$ in A, X_1, X_2 and X_3 in X and B_2 in B) are stated in the first lines which describe the voltage source behavior ($A_{1,1}$ in A, X_1 in X and B_1 in B).

2) Test symbolic expression: A test symbolic expression is then derived from the symbolic matrix expression of the system corresponding to the studied pair (fault/test). This is performed by solving the symbolic matrix expression for the variables involved in the measurement corresponding to the test according to the Cramer's method (see [5]). The resulting test symbolic expression is proven to have a specific multivariable homographic form.

3) Optimization: The uncertainties of the values that can be undertaken by the system parameters is represented by intervals. An algorithm that optimizes the test symbolic expression is used (see [6]). In order to find the corresponding interval outcome of a given test in the occurrence of a given fault, the maximum and the minimum values of the symbolic expression of this test have to be found on the parallelotop defined by the parameters intervals values.

III. TEST SEQUENCING PROBLEM

The test sequencing problem is defined as follows.

- A set F of n_F faults f_i with $i \in \{1, ..., n_F\}$.
- A set p of n_F a priori occurrence probability $p_i \in [0, 1]$ with $i \in \{1, ..., n_F\}$ such that $\sum_{i=1}^{n_F} p_i = 1$.
- A set S of n_S tests s_j with $j \in \{1, ..., n_S\}$. The tests are supposed to be binary, i.e. any test s_j has only two possible outcomes 0 or 1. The tests are also supposed to be symmetrical, i.e. in the occurrence of a given fault f_i, a test s_j has only one possible outcome.
- A set C of n_S test costs $c_j \in \mathbf{R}^+$ with $j \in \{1, ..., n_S\}$ which denote the cost of test s_j measured in term of time, manpower requirements and other economic factors.
- A test-matrix, diagnostic dictionary or "cross-table" $A = [a_{ij}]$ of dimension $n_F \times n_S$ where a_{ij} represents the outcome of the test s_j in the occurrence of the fault f_i ,in this case 0 or 1.

The problem is then to design a test tree (i.e. diagnosis tree) that is able to unambiguously identify the different faults in F using the tests in the test set S, and that minimizes the mean testing cost J given by equation 3 where $d_{ij} = 1$ if test s_j is used in the test sequence leading to the identification of fault f_i and $d_{ij} = 0$ otherwise.

$$J = \sum_{i=1}^{n_F} p_i \times \left(\sum_{j=1}^{n_S} d_{ij} \times c_j \right) \tag{3}$$

It is well-known that the task of finding an optimal diagnosis tree is an NP-complete problem.

In the application AGENDA, tests can be multi-valued (i.e. they can have more than two outcomes) and asymmetrical. Moreover, test costs are not constant but are evaluated dynamically according to the required modifications of the system configuration.

A. *AND/OR search graphs*

The problem of building an optimal diagnosis tree can be formulated as an ordered, best-first search on an AND/OR graph (see [2]). This graph is composed of two kinds of nodes: the OR nodes which are associated to a set of faults to be discriminated (i.e. ambiguity set) and the AND nodes which correspond to the tests to be performed. The root node is an OR node composed of all the anticipated faults (i.e. complete ignorance). One leaf node is an OR node and represents one possible fault (no ambiguity). A non leaf OR node has one and only one AND node child corresponding to the test to be performed. An AND node has several OR node children corresponding to the several possible outcomes of the relative test.

The explicit AND/OR search graph represents all the possible solutions of a given problem starting from the ground elements of this problem. For the test sequencing problem, the ground elements are the fault set, the test set and the corresponding cross-table. Obviously, the possible solutions are all the possible diagnosis trees that allow one to discriminate the faults of the fault set using any subset of the test set.

Because of its too big size, the explicit AND/OR search graph is rarely made explicit. The main idea of the AO* algorithm is to develop only parts of the explicit AND/OR search graph which correspond to the most interesting solutions of the problem, according to the objective function to optimize. For the test sequencing problem, the objective function is the function J to minimize (see equation 3).

This subgraph G, called the implicit AND/OR search graph, is selected according to a relevant Heuristic Evaluation Function (HEF), called h.

The HEF is an easily computable heuristic estimation $h(x)$ of the optimal cost-to-go $h^*(x)$ from any node of ambiguity subset x to the leaf nodes. A HEF h is said to be admissible if and only if $h(x) \leq h^*(x)$ for any ambiguity set x. It has been shown that an admissible HEF h used with the AO* algorithm leads to an optimal diagnosis tree [7]. Moreover, the closer $h(x)$ is to $h^*(x)$, the lower is the dimension of the developed implicit graph G.

At the end of the AO* algorithm, the optimal subgraph G^* is a selected subgraph of the implicit AND/OR search graph G that has been developed. For the test sequencing problem, the optimal subgraph G^* corresponds to the optimal diagnosis tree T^*. For each ambiguity subset x corresponding to the OR nodes that belong to T^*, the cost-to-go value $F(x)$ is equal to the optimal value objective function for this node $J^*(x)$.

B. Admissible HEFs

For the test sequencing problem, two kinds of admissible HEFs have been proposed in the literature. The first one proposed by Yeung in [8], is based on the Shannon's entropy and the second one proposed by Pattipati in [2] on the Huffman's code length. The Yeung's HEF can be used for the specific test sequencing problem that has to be solved whereas the Pattipati's one needs an extension to remain admissible for multi-valued tests. For this purpose, an extension of the Pattipati's heuristic for building D-ary trees (all tests are supposed to have D possible outcomes instead of 2) is proposed by Žužek in [9]. Our contribution improves the heuristic proposed in [2] and in [9] as it allows the test to have a different number of outcomes.

Let x be any ambiguity set composed of n_F^x faults to discriminate f_i^x having occurrence probabilities p_i^x, $i \in \{1, ..., n_F^x\}$.

1) Shannon's entropy based HEF: The Shannon's entropy of x, called $H(x)$, is computed as shown in equation 4.

$$H(x) = -\sum_{i=1}^{n_F^x} p_i^x \times \log_2 p_i^x \tag{4}$$

Let n_j be the number of outcomes corresponding to the test s_j having the cost c_j, $j \in \{1, ..., n_S\}$. Let e_j be the efficiency of the test s_j defined as the maximum decrease of entropy per unit cost when the test s_j is performed(see equation 5).

$$e_j = \frac{\log_2 n_j}{c_j} \tag{5}$$

In order to minimize the cost of the diagnosis tree, the most efficient tests have to be performed first. Assume, without loss of generality, that the tests in S are indexed such that $e_1 \geq e_2 \geq ... \geq e_{n_S}$.

Let ρ be a mapping from $\mathbf{R}^+$ to $\mathbf{R}^+$ defined by equation 6 where $y_r \in \mathbf{R}^+$ with $r \in \mathbf{N}$ is computed as shown in equation 7.

$$\rho(y_r) = \sum_{j=1}^{r} c_j \tag{6}$$

$$y_r = \sum_{j=1}^{r} e_j \times c_j = \sum_{j=1}^{r} \log_2 n_j \tag{7}$$

For a value y such that $y_r \leq y \leq y_{r+1}$, $\rho(y)$ is defined as the interpolation of $\rho(y_r)$ and $\rho(y_{r+1})$ and, hence, is computed as shown in equation 8 where $\gamma(y)$ is the largest value of r such that equation 9 is satisfied.

$$\rho(y) = \sum_{j=1}^{\gamma(y)} c_j + \frac{y - y_{\gamma(y)}}{e_{\gamma(y)+1}} \tag{8}$$

$$y \geq \sum_{j=1}^{r} e_j \times c_j = \sum_{j=1}^{r} \log_2 n_j \tag{9}$$

In [8], it is shown that, for any ambiguity set x, $h^*(x) \geq \rho(H(x))$ where $h^*(x)$ is the optimal cost-to-go from x. Consequently, $\rho(H)$ is an admissible HEF.

2) Huffman's code length based HEF: Let Π be the test sequencing problem that has to be solved. First, let us suppose that all the available tests are binary.

Let Π_2 be the test sequencing problem characterized by a fault set F and a test set S defined as follows.

- F is composed of the n_F considered faults f_i with $i \in \{1, ..., n_F\}$ having an a priori occurrence probability p_i.
- S is composed of the n_S possible symmetrical and binary tests s_j with $j \in \{1, ..., n_S\}$ that can be performed on the fault set F and having an intrinsic constant cost c_j equal to 1.

The test sequencing problem Π_2 is shown to be solved in polynomial time by using the Huffman's algorithm [10].

Since all the possible symmetrical and binary tests are not necessarily available in the test sequencing problem Π, the relation 10 can be established between the optimal solutions $l^*(x)$ and $w_2^*(x)$ of the test sequencing problems Π and Π_2, respectively.

$$w_2^*(x) \leq l^*(x) \tag{10}$$

Assuming, without loss of generality, that the test costs are in ascending order $0 \leq c_1 \leq ... \leq c_{n_S}$, a lower bound $h(x)$ of $h^*(x)$ is obtained by equation 11 where $\lfloor . \rfloor$ represents the floor function (see [2]). h is hence an admissible HEF for the test sequencing problem Π with binary tests.

$$h(x) = \sum_{j=1}^{\lfloor w_2^*(x) \rfloor} c_j + \left([w_2^*(x) - \lfloor w_2^*(x) \rfloor] \times c_{\lfloor w_2^*(x) \rfloor + 1} \right) \tag{11}$$

Let us now suppose that the available tests are multi-valued in the considered test sequencing problem Π. The previous HEF is admissible for the test sequencing problem Π with binary tests but not necessarily with multi-valued tests.

Let Π_M be the test sequencing problem characterized by a fault set F and a test set S defined as follows. Let us assume that the test set S has N_m different available numbers of modalities.

- F is composed of the n_F considered faults f_i with $i \in \{1, ..., n_F\}$ having an a priori occurrence probability p_i.
- S is composed of the N_m subsets of all the possible symmetrical tests having the same number of modalities and an intrinsic constant cost c_j equal to 1. Consequently S is constituted by the n_s tests s_j with $j \in \{1, ..., n_S\}$ coming from the union of these subsets.

A polynomial algorithm that derives an optimal diagnosis tree T_M^* for the problem Π_M from the optimal diagnosis tree T_2^* for the problem Π_2 is proposed (see [11]).

Since all the possible symmetrical tests having the same number of outcomes are not necessarily available in the considered test sequencing problem Π, the relation 12 can be established between the optimal solutions $l^*(x)$ and $w_M^*(x)$ of the test sequencing problems Π and Π_M, respectively.

$$w_M^*(x) \leq l^*(x) \tag{12}$$

Assuming, without loss of generality, that the test costs are in ascending order $0 \leq c_1 \leq ... \leq c_{n_S}$, a lower bound $h(x)$ of $h^*(x)$ is obtained by equation 13 where $\lfloor . \rfloor$ represents the floor function. h is then an admissible HEF for the test sequencing problem Π with multi-valued tests.

$$h(x) = \sum_{j=1}^{\lfloor w_M^*(x) \rfloor} c_j + \Big([w_M^*(x) - \lfloor w_M^*(x) \rfloor] \times c_{\lfloor w_M^*(x) \rfloor + 1} \Big) \tag{13}$$

C. Test set reduction

The AO* algorithm has an exponential complexity depending on the cardinality n_S of the set S of available tests s_j with $j \in \{1, ..., n_S\}$ and the cardinality n_F of the set F of faults f_i with $i \in \{1, ..., n_F\}$ to discriminate.

In the optimal diagnosis tree T^*, let S^* be the subset of the n_{S^*} available tests among the n_S ones which are indeed used to discriminate the n_F faults.

Let τ be the processing time of the AO* algorithm from the available tests set S and τ^* from the available test subset S^*. Since $n_{S^*} < n_S$, $\tau^* < \tau$ since the cross-table has only n_{S^*} columns on the one hand against n_S on the other hand.

So far, there exist no method to select before the execution of the AO* algorithm the available tests s_j which constitute the S^* subset. However, the AO* algorithm complexity can be decreased by reducing the set S of available tests to a subset $S^{'}$ of the most relevant (in term of cost and discriminating power on the F fault set) available tests. However, the diagnosis tree $T^{'}$ resulting from the execution of the AO* algorithm on the selected test subset $S^{'}$ is optimal for the test subset $S^{'}$ itself by definition, but not necessarily for the initial set S. But the optimality loss may be minor compared to the computation gain (see in equation 15).

From a given initial set S of available tests, many test subset $S^{'}$ can be generated. They are characterized according to the three following definitions according to their discriminating power on the set F of the considered faults.

Definition 1 (Discriminating Test Subset) *A test subset $S^{'}$ is said to be discriminating for a fault set F if it is able to discriminate all the faults of the fault set F.*

Definition 2 (Minimal Discriminating Test Subset) *A discriminating test subset $S^{'}$ for the fault set F is said to be minimal if and only if, for any of its $n_S^{'}$ tests $s_j^{'}$ with $j \in \{1, ..., n_S^{'}\}$, $S^{'} - \{s_j^{'}\}$ is not a discriminating test subset for the fault set F.*

Definition 3 (Optimal Discriminating Test Subset) *An optimal discriminating test subset S^* for a fault set F is a discriminating test subset for the fault set F composed of the tests involved in a given optimal diagnosis tree T^* for the fault set F.*

According to the cost function J, the optimality is based on both notions of test cost and discriminating power of the subset of the available tests used in the diagnosis tree. By working exclusively on discriminating test subsets, the notion of discriminating power is always reached. However, it is necessary to be able to evaluate these discriminating test subsets in order to select the one which is the most likely to provide the diagnosis tree that has the lowest cost value J.

Obviously, the ideal evaluation of a discriminating test subset $S^{'}$ would be proportional to the cost $J(T^{'})$ associated to the optimal diagnosis tree $T^{'}$ obtained by executing the AO* algorithm. However, it is impossible to predict the cost value $J(T^{'})$ from the $S^{'}$ test subset without knowing the diagnosis tree $T^{'}$ itself.

As shown in equation 14, the heuristic evaluation function K of a test subset $S^{'}$ is defined as the sum of the costs c_j corresponding to the tests s_j that belong to this subset $S^{'}$.

$$K(S^{'}) = \sum_{j=1 | s_j \in S^{'}}^{n_S} c_j \tag{14}$$

Whatever the selected heuristic function K used to evaluate the cost of the test subsets, finding the lowest K value discriminating test subset is always a NP-complete problem. Moreover, the optimal solution of this problem is not necessarily an optimal discriminating test subset S^*. So, it seems reasonable to use a polynomial algorithm that provides only a suboptimal solution for this problem.

A two step discriminating test subset generation polynomial algorithm is hence proposed. The first step consists in selecting the tests of the initial test set S one by one according to the lowest cost value until a first discriminating test subset, called S'_{first}, is obtained. The second step generates from the previously obtained first discriminating test subset S'_{first}, a minimal discriminating test subset, called S'_{min}.

IV. PERFORMANCE EVALUATION

The quality criterion Q, defined as shown on equation 15, illustrates the interest of a discriminating test subset S' as the gain in term of AO* algorithm processing time versus the loss in term of optimality (τ'_{min} represents the processing time of the AO* algorithm from the minimal discriminating test subset S'_{min}).

$$Q(S') = \frac{J(T^*)}{J(T')} \times \frac{Min(\tau^*, \tau'_{min})}{\tau'} \qquad (15)$$

The performance has been evaluated on real electronic circuits of the automotive domain. From a given test subset S', the AO* algorithm with the admissible Yeung's HEF returns the same optimal value $J(T')$ with the same magnitude order of processing time τ' as with the admissible Pattipati's HEF.

For all the evaluated electronic circuits, the best $Q(S')$ criterion value is always obtained for S'_{min} or S^* and the worst one for S. It is also interesting to remark that, for some circuits, the $Q(S')$ criterion value is almost the same for S'_{first} as for S'_{min}.

Consequently, since the optimal discriminating test subset can not be obtained a priori, the selection of a minimal or even a first discriminating test subset seems to be an interesting trade-off between the gain in AO* algorithm processing time and the optimality loss.

V. FUTURE WORKS

Two major issues that are considered for the future are developed in this section.

The first one deals with the reduction of the test set in order to compute diagnosis trees quicker and with a lower complexity. Building an optimal diagnosis tree for electronic circuits is an NP-complete problem. Its complexity is linked with the number of tests and the number of faults. In order to reduce this complexity we search a method to select a discriminant subset of tests which can be used to compute the diagnosis tree. This selection is made according to a criteria taking into account the dynamic cost and the number of modalities for a test. It is important to notice that the optimality is not guaranteed but an upper bound of the cost should be a priori estimated (see [8]). We are considering to base our work on [12] whose method could be applied before the prediction algorithm. The principle is to consider the behavioral model to find discriminant test subsets. Then from the different obtained subsets, by applying the criteria, we hope to have the subset which will give the tree which has the nearest cost of the optimal one.

The second issue concerns the diagnosis tree traverse taking into account the exact values of the already performed tests. During a diagnosis session, exact values are measured by the garage mechanics and could be stored. On the other hand, thanks to the prediction algorithm, we have all the formal expressions for each available test. The main idea is to use this equation and the exact measured values to compute again the bounds of the interval or better an exact value for not yet used tests. So we add more precision in the numeric values used to fill the cross-table and consequently the tests become more and more pertinent. It is obvious that computation must be light here and based on all what we have computed off-line because all these changes will have to be done on-line.

REFERENCES

[1] L. Travé-Massuyès, P. Dague, and F. Guerrin, *Le raisonnement qualitatif.* HERMES, 1997.

[2] K. R. Pattipati and M. G. Alexandridis, "Application of Heuristic Search and Information Theory to Sequential Fault Diagnosis," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 20, pp. 872–887, July 1990.

[3] P. Dague, O. Raiman, and P. Devès, "Troubleshooting : when modeling is the trouble," in *AAAI*, (Seattle (USA)), pp. 600–605, 1987.

[4] A. Darwiche, "Model-Based Diagnosis using Structured System Descriptions," *Artificial Intelligence*, vol. 8, pp. 165–222, 1998.

[5] M. E. Kole, *Algorithms for Symbolic Circuit Analysis Based on Determinant Calculations.* PhD thesis, Universiteit Twente, 1996.

[6] S. Group, "Modal Intervals : draft for a basic tutorial," tech. rep., IMA Informática y Matemática Aplicada department, University of Girona, Oct. 1998.

[7] A. Bagchi and A. Mahanti, "Admissible Heuristic Search in AND/OR Graphs," *Theoritical Computer Science*, vol. 24, pp. 207–219, 1983.

[8] R. W. Yeung, "On Noiseless Diagnosis," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 24, no. 7, pp. 1074–1082, 1994.

[9] A. Žužek, A. Biasizzo, and F. Novak, "Sequential diagnosis tool," *Microprocessor and Microsystems*, vol. 24, pp. 191–197, 2000.

[10] D. A. Huffman, "A method of the construction of minimum redundancy codes," *Proceedings IRE*, vol. 40, pp. 1090–1101, 1962.

[11] P. P. Faure, L. Travé-Massuyès, and H. Poulard, "An Interval Model-Based Approach for Optimal Diagnosis Tree Generation," in *10th International Workshop on Principles of Diagnosis, DX-99*, (Loch Awe (Scotland)), pp. 78–89, 1999.

[12] L. Travé-Massuyès, T. Escobet, and R. Milne, "Model-Based Diagnosability and Sensor Placement - Application to a Frame 6 Gas Turbine Subsytem," in *12nd International Workshop on Principles of Diagnosis, DX-01*, 2001.