Available online at www.sciencedirect.com

ScienceDirect

IFAC-PapersOnLine 49-11 (2016) 555–561

Combining Multiple Diagnostic Trouble Codes into a Single Decision Tree

Markos F. B. G. Oliveira* Ricardo Lüders*

* *Federal University of Technology - Paraná, Curitiba, 80230-901 PR Brazil (e-mail: markos@hotmail.com, luders@utfpr.edu.br).*

Abstract: Recent advances in design of automotive systems have yielded to vehicles that now include several complex electronic devices. They are able to log hundreds of diagnostic trouble codes (DTC) for different malfunctioning conditions with a corresponding fault tree built for each code. However, many of these active codes can have common causes which make diagnostic a very complex task. Using binary decision diagrams (BDD), we propose a strategy for combining individual BDDs developed for each DTC into a single one. This strategy takes into account sets of both active and non-active DTCs representing the current diagnostic scenario. The results show a significant decrease in the total diagnostic cost.

Keywords: Fault diagnosis, fault tree analysis, binary decision diagram, automotive system.

1. INTRODUCTION

In the last years there was a significant improvement in diagnostic of automotive systems by providing vehicles with complex electronic devices (Suwatthikul, 2010). Embedded electronic systems are now able to store lots of diagnostic information as Diagnostic Trouble Codes (DTCs) generated by faulty components. These codes are triggered when a system fail occurs. During maintenance or repair, DTCs can be recovered from internal vehicle's memory by specialized technicians using guided diagnostic tools. However, many DTCs are triggered due to a cascade effect caused by common faulty components making diagnostic a burden and time consuming task. For example, an electrical fault caused by a short-circuit to ground in a cable connecting an electronic control unit (ECU) to a temperature sensor can trigger the DTCs for both ECU and sensor.

The diagnostic of such systems are based on fault trees analysis (FTA) built by diagnostic specialists. Fault trees are often partitioned into vehicles' subsystems and maintained by even different teams. One fault tree is built for each DTC by associating causes (faulty components) to it and then encoded into a diagnostic decision tree (DTC) to implement sequences of component tests.

The conversion process of a fault tree to the corresponding DTC is discussed in Rauzy (1993); Reay and Andrews (2002); Remenyte and Andrews (2006) using a binary decision diagram (BDD). This process requires a particular ordering of basic events. In our work, this ordering is provided by the cost of diagnostic importance factor (CDIF) introduced by Assaf and Dugan (2008). Among several indicators found in the literature for assessing risk significance, the diagnostic importance factor (DIF) measures the probability of a particular basic event be the cause of the system fault (Dutuit and Rauzy, 2013). Thus it provides a natural measure to define which component should be tested first.

In this work, we propose a diagnostic strategy that builds a single BDD by combining multiple BDDs of active and non-active DTCs. Therefore, current BDDs already developed for each DTC can be used, preserving diagnostic efforts made previously by industry. In the end, unnecessary tests are avoided based on the current diagnostic scenario. A similar idea can be found in Yuan and Hu (2009), but it does not consider any measure of test importance or diagnostic cost. Another related work can be found in Assaf and Dugan (2008) where evidences provided by monitors and sensors are incorporated into diagnostic. A diagnostic cost proposed by the authors is used to show that a less complex diagnostic is then obtained. However, they do not consider multiple FTs. The advantage of our strategy is to reduce the diagnostic cost while dealing with multiple faults. This is shown by using an illustrative example, although it is not replace a real-world study case to be considered in the future.

This paper is organized as follows. Section 2 presents a background on fault tree analysis. Section 3 describes the problem of dealing with multiple fault codes. Section 4 presents our strategy for combining multiple BDDs and Section 5 presents an illustrative example by comparing diagnostic costs obtained for different approaches. Concluding remarks are presented in Section 6.

2. PRELIMINARIES

Most material presented in this section concerns building a BDD that encodes the minimal solutions of a boolean function as presented in Rauzy (1993). Moreover, DIF and CDIF are defined according to Dutuit and Rauzy (2013) and Assaf and Dugan (2008), respectively.

Example 1. A fault tree is a graphical representation of (logical) combinations of basic events (or components)

⋆ This work is supported by the undergraduate research program of PROPPG/UTFPR. A long term support is also recognized from the Brazilian agencies CNPq, CAPES and Fundação Araucária.

10.1016/j.ifacol.2016.08.081

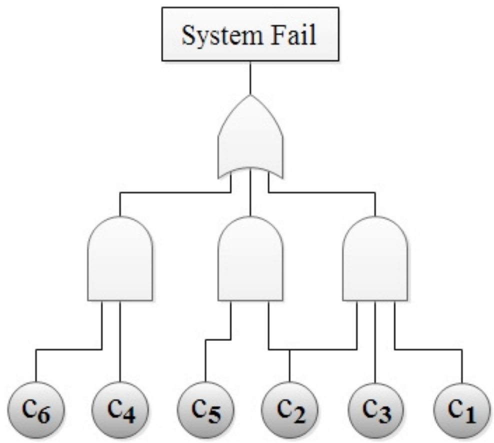

Fig. 1. Fault tree for Example 1.

that can be faulty which generates a system fail s as depicted in Fig. 1. Suppose a system with components $c_1, \ldots, c_6$ that can be faulty which generates an observed system fail. A component is set to 1 when it is faulty and 0 otherwise. Similarly s is set to 1 (active) under system fail and 0 otherwise. The relationship between system fail and faulty components is given by the boolean expression (1).

$$s = c_6c_4 + c_5c_2 + c_2c_3c_1 \tag{1}$$

A strategy of diagnostic is then a sequence of tests (pass or failed) to identify faulty components causing a system fail.

2.1 Minimal cutsets

Boolean variables are denoted by letters $c, c_1, c_2, \ldots$, while $s, s_1, s_2, \ldots$ denote boolean expressions. Variables occurring in a boolean expression s are given by $\text{var}(s)$.

Definition 2. (Boolean expression). A boolean expression over an enumerable set $\mathcal{C}$ of variables (components or basic events) is recursively defined by (i) a variable is a boolean expression; (ii) if s_1 and s_2 are boolean expressions, then $s_1 + s_2$, $s_1 \cdot s_2$ and $\overline{s}_1$ are boolean expressions as well.

Definition 3. (Variable assignment). An assignment σ of $\text{var}(s) \in \mathcal{C}$ is a function from $\text{var}(s)$ into $\{0, 1\}$.

Definition 4. (Boolean function). A boolean function of n variables is a mapping $f : \{0, 1\}^n \to \{0, 1\}$.

Definition 5. (Minterm). A minterm is a product of variables c or $\overline{c}$ (not both) for all $c \in \text{var}(s)$.

There is a one-to-one mapping between variable assignments and minterms. Given an assignment σ and a minterm π, $c \in \text{var}(\pi)$ if $\sigma(c) = 1$ and $\overline{c} \in \text{var}(\pi)$ if $\sigma(c) = 0$. Therefore, σ or π are used interchangeable to denote variable assignments or minterms.

Similarly, there is a unique boolean function corresponding to a given boolean expression (not the opposite). Therefore, we refer interchangeable to s as a boolean expression or function according to convenience.

A partial order is defined in the set of minterms built over $\mathcal{C}$ (and consequently in the set of variable assignments of $\mathcal{C}$). Given two minterms σ and π and $c \in \mathcal{C}$, $\sigma \preceq \pi$ iff $c \in \sigma$ then $c \in \pi$.

Definition 6. (Coherent boolean function). A function s is coherent if for any two minterms σ and π such that $\sigma \preceq \pi$, $\sigma \in s$ implies that $\pi \in s$.

In other words, faulty components that cause a system fail must cause fail for all other component conditions.

Given an expression s, $s|c_1, \ldots, c_k$ is the boolean expression evaluated for $c_1 = \cdots = c_k = 1$ and $\overline{c}_1 = \cdots = \overline{c}_k = 0$ in s.

Proposition 7. (Logical Shannon decomposition). Let s be a boolean expression and $c \in \text{var}(s)$. Then,

$$s \equiv c \cdot (s|c) + \overline{c} \cdot (s|\overline{c}) \tag{2}$$

Proposition 8. (Probabilistic Shannon decomposition). Let s be a boolean expression and $c \in \text{var}(s)$. Then,

$$\begin{aligned} \Pr\{s\} &= \Pr\{c\} \cdot \Pr\{s|c\} + (1 - \Pr\{c\}) \cdot \Pr\{s|\overline{c}\} \\ &= \Pr\{c\} \cdot (\Pr\{s|c\} - \Pr\{s|\overline{c}\}) + \Pr\{s|\overline{c}\} \end{aligned} \tag{3}$$

Let π be a positive product of boolean variables of $\mathcal{C}$. The minterm obtained by completing π with negative literals of $\mathcal{C}$ that do not appear in π is denoted by $\lfloor \pi \rfloor_{\mathcal{C}}$.

Definition 9. (Minimal cutset). Let s be a boolean function and π be a product of boolean variables of $\text{var}(s)$. Then π is a cutset of s if $\lfloor \pi \rfloor_{\text{var}(s)} \in s$. In addition, π is a minimal cutset of s if it is a cutset of s and there is no cutset σ of s such that $\sigma \prec \pi$.

Proposition 10. Let s be a boolean function and $\text{MCS}(s)$ be the disjunction of minimal cutsets of s. Then s is coherent iff $\text{MCS}(s) \equiv s$.

2.2 DIF and CDIF

DIF measures the importance of a component for diagnostic, i.e. it is the probability of a particular component is faulty given the system is faulty.

Definition 11. (DIF). Given a boolean function s and a component $c \in \mathcal{C}$, the diagnostic importance factor $\text{DIF}(s, c)$ is given by

$$\text{DIF}(s, c) \doteq \Pr\{c|s\} = \frac{\Pr\{c \cdot s\}}{\Pr\{s\}} \tag{4}$$

It measures the risk of system fail due to a particular component fault c. It can be approximated by (5) and (6) according to Dutuit and Rauzy (2013).

$$\text{DIF}(s, c) \approx \frac{\Pr\{c\} \cdot \Pr\{s|c\}}{\sum_{\pi \in \text{MCS}(s)} \Pr\{\pi\}} \tag{5}$$

with

$$\Pr\{s|c\} \approx \sum_{c \cdot \pi \in \text{MCS}(s)} \Pr\{\pi\} + \sum_{\pi \in \text{MCS}(s), c \notin \pi} \Pr\{\pi\} \tag{6}$$

Assaf and Dugan (2008) have introduced CDIF. It includes a cost for component test as, for instance, time required for testing or test complexity.

Definition 12. (CDIF). Given a boolean function s and a component $c \in \mathcal{C}$ whose cost of testing is t_c, the cost of importance factor $\text{CDIF}(s, c)$ is given by

$$\text{CDIF}(s, c) = \frac{\text{DIF}(s, c)}{t_c} \tag{7}$$