British Journal of Mathematics & Computer Science
10(2): 1-9, 2015, Article no.BJMCS.18255
ISSN: 2231-0851

SCIENCEDOMAIN *international*
www.sciencedomain.org

Topological Conditions for Automated Diagnostic Software to Estimate the Diagnosability of Digital Devices at the Structural Level

Rami Matarneh[1*]

[1]*Department of Computer Science, University of Hail, Hail, Saudi Arabia.*

Article Information

DOI: 10.9734/BJMCS/2015/18255
Editor(s):
(1) Feyzi Basar, Department of Mathematics, Fatih University, Turkey.
(2) Tian-Xiao He, Department of Mathematics and Computer Science, Illinois Wesleyan University, USA.
Reviewers:
(1) Anonymous, Sastra University, India.
(2) F. W. S. Lima, Departamento de Física, Universidade Federal do Piauí, Teresina, Brazil.
(3) Anonymous, Punjab Technical University, India.
Complete Peer review History: http://sciencedomain.org/review-history/9921

Original Research Article

Received: 11 April 2015
Accepted: 18 May 2015
Published: 27 June 2015

Abstract

Diagnosability is the property of a partially observable system with a given set of possible faults; these faults can be detected with certainty with a finite observation. Usually, the definition and the verification methods of diagnosability ignore the nature of controllable and uncontrollable events of the system. This paper shows the influence of controllability of system's events on the definition and the verification, also shows that the classical diagnosability is a special case where we consider the whole system as controllable. The definition of diagnosability had been generalized using model structure on topological spaces by mean of strategies. Alternating-time Temporal Logic and Model Checking are used to check diagnosability of uncontrollable events to build a whole framework which is suitable for both isolated and interacting systems.

Keywords: Topological conditions; automated diagnostic; diagnosability; temporal logic.

1 Introduction

At various stages of the design of computer hardware it is advisable to maintain the general requirements for the diagnosis, which is mean the capability to conduct the process of finding defects and faults by using specific technical diagnostics based on well-defined methods. Diagnosable digital device can be provided with specific hardware and time costs during the design of diagnostic software and during the diagnostic experiment.

**Corresponding author: ramimatarneh@gmail.com;*

Time costs take into consideration the complexity of performing certain operations, and the characteristics of the hardware to ensure the diagnosis [1,2]. The most appropriate quantitative indicators such as functional logic and circuit realization are diagnosed at key stages of the system using the top-down design.

At the structural level, the quantitative estimation is comprised of overhead and the significance of individual design in the verification process of their functionality, so, diagnosis of digital devices at this level can be achieved by improving the reachability of the test's source by adding additional control points.

To calculate the testability indicators of individual design we can use table of functionality, which allows us to automate and simplify the calculation process [3,4], this approach will make the results more accurate due to its dependence on both controllability and observability.

2 Topological Condition of Diagnostic Algorithms

Let the structure of the digital device is represented as a directed graph $G = (V, E)$ with a number of nodes $0, 1 , 2, \ldots , v$ and a number of arcs $1, 2, \ldots , E$. We assume that the design is composed of an internal or external hardware core capable of generating test inputs then collect and analyze the test outputs.

At the beginning we specify the first node of the graph with subscript V_0, and call it base-node [5,6]. To ensure the diagnosis process the base-node must be able to reach each node of the graph by means of active control points which will be fed by the inputs of the test. On contrast the outputs in response to the inputs will be collected by another type of control points, which will be called passive control points; in the meantime, it shall be ensured reverse-reachability to any node of the graph from the passive control points. The concepts of reachability and reverse-reachability can be illustrated by the use of appropriate matrices. Reachability matrix $R = [r_{ij}]$ describes all the possible paths from V_i node to V_j node. Accordingly, the element of matrix R will be:

$$r_{ij} = \begin{cases} 1, \textit{ if there is a path from } v_i \textit{ to } v_j, \\ 0, \textit{ otherwise.} \end{cases} \tag{1}$$

The set of nodes $R(v_i)$ of graph Gwhich can be reached from a given node $v_i$is consisting of elements v_j for which path L_{ij} between $v_i$and v_j in the reachability matrix is equal to 1 [7,8,9]. It is obvious that all the diagonal elements of matrix $R$are equal to 1, since each node is reachable from itself with a path length equals to 0.

Let $T_1(v_i)$ is the set of nodes of v_j j, which are reachable from v_i with path of length 1, $T_2(v_i)$ - the set of nodes that are reachable from v_i with path of length 2. Similarly, $T_k(v_i)$ is the set of nodes that are reachable from $v_i$paths of length K, in general, the nodes that are reachable from $v_i$can be summarized as follows:

$$R(v_i) = \{v_i\} \cup \{T_1(v_i)\} \cup \{T_2(v_i)\} \cup \ldots \cup \{T_k(v_i)\} \tag{2}$$

Similarly, we define the elements of reverse-reachability matrix $Q(v_i) = [q_{ij}]$ as follows:

$$q_{ij} = \begin{cases} 1, \textit{ if from node } v_i \textit{ we can reach } v_j, \\ 0, \textit{ otherwise.} \end{cases}$$

Reverse-reachability matrix $Q(V_i)$ of graph G can be described as a set of nodes, such that, from any of its nodes we can reach the node v_i. It is clear that columnv_i of matrix Qcoincides with the line $v_i$of matrix R, i.e.$Q = R^T$, where R^T- transposed matrix of the reachability matrix R. At the given matrices R and Q we can impose some restrictions, for example, let the lengths of the paths do not exceed a specified number α, then the matrices R^α and Q^α will be respectively called the constraint matrix of α for both reachability and reverse-reachability matrices. For further discussion let us use the concepts: base and anti-base of graph theory [10,11,12], where base nodes will match the set of input nodes, and anti-base will match the set of output nodes.

Definition 1. The base B of graph G is a set of nodes $v_1, v_2, \ldots, v_i$, such that, it can reach any node in the graph, and seems to be minimal in the sense that there is no proper subset of B has the same characteristic. Let $R(B)$ the set of nodes that are reachable from the nodes of set B, this we can rewrite in the following form:

$$R(B) = \underset{V_i \in B}{U} R(V_i). \qquad (3)$$

Approval. B seem to be base node if and only if

$$R(B) = v \; and \; (\forall S \subset \mathrm{B}\big((\mathrm{R(S)} \neq \mathrm{V})\big) \qquad (4)$$

$\mathrm{R(S)} \neq \mathrm{V} \subset \forall S \subset \mathrm{B}$ in (4) is equivalent to the following statement: $v_i \not\in (R(v_i))$ for any two different $v_i, v_j \in B$, i.e., no node in B can be reached from another node in B.

Thus, the set of base node $B$of graph G, must satisfies the following conditions:

1. Each node in graph G is reachable from at least one node of the base set B.
2. No node in set B is reachable from another node of B.

Definition 2. Anti-base $\bar{B}$ is a set of nodes of G such that:

$$Q(\overline{B}) = \underset{V_i \in B}{U} Q(V_i) = V \quad \text{and} \quad \forall S \subseteq \overline{B},\; Q(S) = V \qquad (5)$$

Therefore, anti-base $\bar{B}$ is the minimum possible set of nodes of graph G, such that, some vertices in B are reachable through it. Using the concepts of base and anti-base, we can formulate a method for selecting the points of diagnostics as follows: determine a minimum number N of nodes for inputs and outputs without any restrictions on test time T_{test}. For a finite strongly connected directed graph $G(V, E)$ find:

$$N = \min\big(B_r(G)\bar{B}_s(G)\big) \qquad (6)$$

Where $B_r(G)$ and $\bar{B}_s(G)$ - one of base and anti-base sets.

But regarding the process of determining the base and anti-base is quite simple [9,13,14]. Here are two basic rules for choosing the base set.

1. In any graph without cycles, there exists a unique base; it consists of all nodes of the graph, whose indegree is 0.
2. For a graph G with cycles, it is necessary to determine the strong components, then construct the graph $G^*(V^*, E^*)$, so that, each node of them represents a set of nodes of the strong component.

Graph G^* is called the condensation of graph G which does not contain cycles and base B^* is a condensation of G^* consisting of all nodes with indegree equal to 0. Consequently, the base graph Gcan be constructed as follows: the corresponding nodes of B^* to strong components of Gmust be taken one-by-one, i.e. Assume $B^* = \{S_1, S_2, \ldots, S_m\}$, where m the number of sets nodes S_i in graphB^*, then the base B is an arbitrary set $\{V_{i1}, V_{i2}, \ldots, V_{im}\}$, where $V_{ij} \in S_j$. Searching for a strong component of a graph containing the nodeV_i, can be carried out as follows:

Since $R(V_i)$ is the set of nodes reachable from $V_i$and $Q(V_i)$ is the set of nodes from which we can reach V_i, then $R(V_i) \cap Q(V_i)$ uniquely identifies a strong component with respect to node V_i. When taking into consideration the relationship between the matrix R^T and Q, we find that search for a strong component with respect to each node $V_i(i = 1, V)$ we have to perform elementwise logical multiplication of similar rows and columns of the reachability matrix R. After multiplying all rows and columns we will get all the strong components of the graph, where some of them may coincide with others if they have the same nodes name [15,16,17,18].

To check the reality and the effectiveness of the mentioned method, let us take Fig. 1 as an example. The structure has 13 nodes, there is unidirectional and bidirectional connection between the nodes, which is simulate a real unidirectional and bidirectional buses for information transfer.

Firstly, we build reachability matrix $R$of the graph as shown in Table 1.

Secondly, performing elementwise logical multiplication of similar rows and columns of the matrix, we obtain a set of nodes for each strong component of the original graph G.

Thirdly: Denote the set of nodes for each V_i^*: $V_1^*\{v_1, v_2, v_5, v_6\}$, $V_2^*\{v_3\}$, $V_3^*\{v_4, v_7, v_9\}$, $V_4^*\{v_8, v_{10}\}$, $V_5^*\{v_{11}, v_{12}, v_{13}\}$.

The nodes of B^* with indegree equal to 0, areV_2^*, V_5^* respectively, and corresponding to the sets $\{V_3\}$, $\{V_{11}, V_{12}, V_{13}\}$ As shown in Fig. 2.

Thus, the bases of the original graph$G$are the following pairs: $\{V_3, V_{11}\}$, $\{V_3, V_{12}\}$ and $\{V_3, V_{13}\}$where the anti-base nodes of a graph G will be $\{V_4\}$,$\{V_7\}$ and $\{V_9\}$. Any of bases can be used as an active control points while anti-bases can be used as passive control points.

Since the base and the anti-base of the original graph Ghave no common nodes, the minimum number of connection points of diagnosis is obtained by combining arbitrarily chosen base and anti-base of graph G. Actually the number of permutation of the points here is all possible pairs of three bases and three anti-bases:

$$\{V_3, V_{11}, V_4\}, \{V_3, V_{11}, V_7\}, \{V_3, V_{11}, V_9\}, \dots, \{V_3, V_{13}, V_9\}$$

Connecting diagnostic software tools through active and passive control points requires certain hardware costs for additional circuits such as switches, buffer amplifiers, circuits… etc.

Table 1. Reachability matrix $R$of graph G

V_i	1	2	3	4	5	6	7	8	9	10	11	12	13
1	1	1	0	1	1	1	1	1	1	1	0	0	0
2	1	1	0	1	1	1	1	1	1	1	0	0	0
3	1	1	1	1	1	1	1	1	1	1	0	0	0
4	0	0	0	1	0	0	1	0	1	0	0	0	0
5	1	1	0	1	1	1	1	1	1	1	0	0	0
6	1	1	0	1	1	1	1	1	1	1	0	0	0
7	0	0	0	1	0	0	1	0	1	0	0	0	0
8	0	0	0	1	0	0	1	1	1	1	0	0	0
9	0	0	0	1	0	0	1	0	1	0	0	0	0
10	0	0	0	1	0	0	1	1	1	1	0	0	0
11	0	0	0	1	0	0	1	1	1	1	1	1	1
12	0	0	0	1	0	0	1	1	1	1	1	1	1
13	0	0	0	1	0	0	1	1	1	1	1	1	1

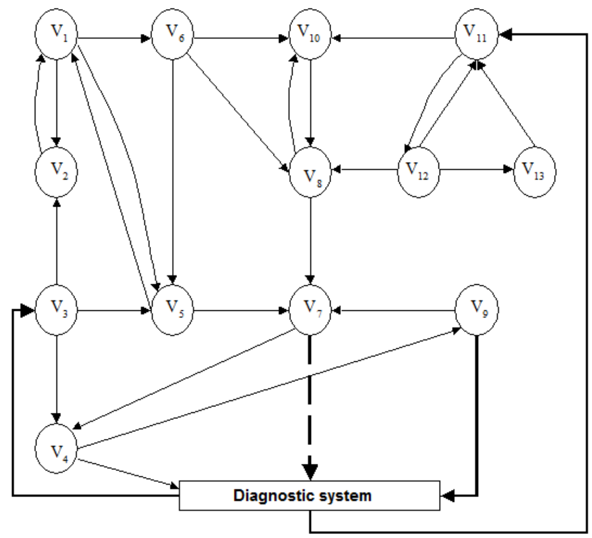

Fig. 1. Base graph *G*

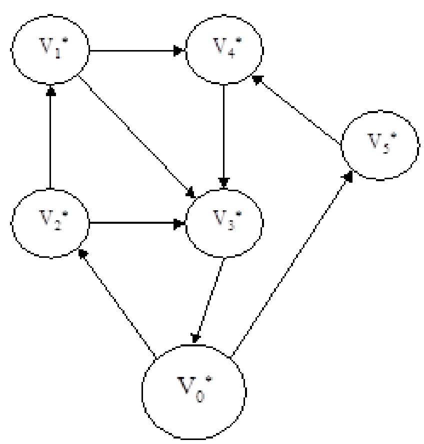

Fig. 2. Condensation of graph *G*

Because of high importance of evaluation and optimization of hardware costs for diagnosis [19,20,21,22]; we have to specify the indicators that characterize such costs when starting diagnosis process.

C_i– Minimum cost to connect diagnostic software to active control point, where $i = 1, \ldots, m$, and m represents the number of active control point.

C_j– Minimum circuit cost to connect passive control point, where $j = 1, \ldots, n$ and n represents the number of passive control point.

The total cost can be represented as follows:

$$C = \sum_{i=1}^{m} C_i + \sum_{j=1}^{n} C_j \tag{7}$$

Taking into consideration that active points represent base nodes and passive points represent anti-base nodes, we can express the hardware cost by using the characteristic function $\mu(V)$ where its value will show whether V belongs to base B or anti-base $\bar{B}$:

Thus, the total cost will equal to:

$$\mu(V) = \begin{cases} 1, \text{if } V \in B, \\ 0, \text{if } V \notin B. \end{cases} \quad (8)$$

Where

$$\mu_{ir}(V) = \begin{cases} 1, \text{if } V \in B_r; \\ 0, \text{ if } V \notin B_r \end{cases}$$

$$C = \sum_{i=1}^{|V|} C_{ir}\mu_{ir}(V) + \sum_{j=1}^{|V|} C_{js}\mu_{js}(V) \quad (9)$$

B_r- is one of the total number of bases R in graph G, where $r = 1, 2, \dots, R$.

$$\mu_{js}(V) = \begin{cases} 1, \text{if } V_j \in B_S \\ 0, \text{if } V_j \notin B_S. \end{cases}$$

B_s- is one of the total number of anti-bases S in graph G, where $s = 1, 2, \dots, S, B_s \in S$.

Similarly, we can estimate the required time for the diagnosis, which can be determined by applying inputs stimuli to active control points and collecting outputs on passive control points.

If we assume that each path L is checked sequentially and if we connect one active and one passive control point to each other, then the total diagnosis time T_g is equal to:

Where

$$T_g = \sum_{k=1}^{|V|} \sum_{l=1}^{L} \mu_{kl}(V)\, t_k \quad (10)$$

L - Total number of paths between control points;

$$\mu_{kl}(V) = \begin{cases} 1, \text{if } V_k \text{ belongs to } 1^{\text{st}} \text{ path;} \\ 0, \text{otherwise.} \end{cases}$$

t_k – Time needed to test one node of the graph.

The total time T_g depends on the choice of the set of paths L of the graph For each path of a certain length, we can specify its allowable scan time $T_l = (1, 2, \dots, L)$.

So, the total time of diagnosis will be equal to the sum of these times:

$$T_g = \sum_{l=1}^{L} T_l$$

Based on the above given relations for the evaluation of hardware and time costs, we can formulate two tasks to optimize data cost.

1. Task to minimizing the hardware requirements for diagnosis. Find

Limitations of this task can have two different types depending on the technical conditions. If we set the total allowable time of diagnosis T_g, then the limitations will be:

$$\min C = \sum_{i=1}^{|v|} C_{ir}\mu_{ir}(v) + \sum_{j=1}^{|v|} C_{js}\mu_{js}(v) \tag{11}$$

$$\sum_{k=1}^{|v|}\sum_{l=1}^{L} \mu_{kl}(v)\, \mathrm{t_k} \le T_g . \tag{12}$$

Limits of allowable scan time of T_l for each l path can be written as

The optimal values of this task $t_{k,} C_{ir}, C_{js} \ge 0$.

$$\sum_{k=1}^{|v|} \mu_{kl}(v)\, T_k \le T_1 \quad (l = 1,\ 2,\ \dots,\ L). \tag{13}$$

2. The second task deals with minimizing the required time for diagnosis which can be stated as follows:

Find

$$\min T_g = \sum_{k=1}^{|v|}\sum_{l=1}^{L} \mu_{kl}(v)\, t_k . \tag{14}$$

Under the constraints:

$$\begin{cases} \sum_{i=1}^{|v|} C_{ir}\mu_{ir}(v) \le C_1, \\ \sum_{j=1}^{|v|} C_{is}\mu_{is}(v) \le C_2. \end{cases} \tag{15}$$

Where C_1, C_2 – the allowable hardware costs to connect active C_1 and passive C_2 control points to the diagnostic software, taking into consideration that $C_1, C_2 \ge 0$.

3 Conclusions

Using the characteristic function μ(V) allowed us to have limits on the values of the sum of identical and equal nodes to the total number of nodes of the graph, which is more convenient for solving optimization problems in the future. As for the particular graph there are a lot of bases and anti-bases then the choice of the pair "basis – anti-base" can be determined by specific technical conditions depending on hardware cost with relation to diagnostic software.

Competing Interests

Author has declared that no competing interests exist.

References

[1] Caruso Antonio, et al. Diagnosability of regular systems. Journal of Algorithms. 2002;45(2):126-143.

[2] Allan FJ, Tiko Kameda, Toida S. An approach to the diagnosability analysis of a system. IEEE Transactions on Computers. 1975;24(10):1040-1042.

[3] Ehling E, Jackson P, Mccarthy J. Automated diagnostic testing system. U.S. Patent No. 3,854,125; 1974.

[4] English Roe E, Michael A. Kilgore, Jerry A. Crone. Automated diagnostic system. U.S. Patent No. 4,967,337; 1990.

[5] Barsi Ferruccio, Fabrizio Grandoni, Piero Maestrini. A theory of diagnosability of digital systems. Computers, IEEE Transactions. 1976;100(6):585-593.

[6] Pajic Miroslav, et al. Topological conditions for wireless control networks. Decision and control and European control conference (CDC-ECC), 2011 50th IEEE conference on. IEEE; 2011.

[7] Alfred V. Aho, John E. Hopcroft. The design and analysis of computer algorithms. Addison-Wesley Longman Publishing Co., Inc., Boston, MA; 1974.

[8] Contant Olivier, Stéphane Lafortune, Demosthenis Teneketzis. Diagnosability of discrete event systems with modular structure. Discrete Event Dynamic Systems. 2006;16(1):9-37.

[9] Thorsley David, Demosthenis Teneketzis. Diagnosability of stochastic discrete-event systems. Automatic Control, IEEE Transactions. 200550(4):476-492.

[10] Iliff Edwin C. Automated diagnostic system and method including multiple diagnostic modes. U.S. Patent No. 6,730,027; 2004.

[11] Lai Pao-Lien, et al. Conditional diagnosability measures for large multiprocessor systems. Computers, IEEE Transactions. 2005;54(2):165-175.

[12] Lai Pao-Lien, et al. The diagnosability of the matching composition network under the comparison diagnosis model. Computers, IEEE Transactions. 2004;53(8):1064-1069.

[13] Milchtaich Igal. Topological conditions for uniqueness of equilibrium in networks. Mathematics of Operations Research. 2005;30(1):225-244.

[14] Ozawa Takao. Topological conditions for the solvability of linear active networks. International Journal of Circuit Theory and Applications. 1976;4(2):125-136.

[15] Pecheur Charles, Alessandro Cimatti, Ro Cimatti. Formal verification of diagnosability via symbolic model checking. Workshop on Model Checking and Artificial Intelligence (MoChArt-2002), Lyon, France; 2002.

[16] Somani Arun K, Vinod K. Agarwal, David Avis. On the complexity of single fault set diagnosability and diagnosis problems. Computers, IEEE Transactions. 1989;38(2):195-201.

[17] Jiang Shengbing, et al. A polynomial algorithm for testing diagnosability of discrete-event systems. Automatic Control, IEEE Transactions. 2001;46(8):1318-1321.

[18] Wong Joseph D, Peter A. Put. Systems and methods for providing an automated diagnostic audit for cluster computer systems. U.S. Patent No. 6,836,750; 2004.

[19] Sullivan Gregory. A polynomial time algorithm for fault diagnosability. 2013 IEEE 54th Annual Symposium on Foundations of Computer Science. IEEE; 1984.

[20] Ding Yu, Jianjun Shi, Dariusz Ceglarek. Diagnosability analysis of multi-station manufacturing processes. Journal of Dynamic Systems, Measurement and Control. 2002;124(1):1-13.

[21] Übeylı Elif Derya, İnan Güler. Feature extraction from Doppler ultrasound signals for automated diagnostic systems. Computers in Biology and Medicine. 2005;35(9):735-764.

[22] Travé-Massuyes, Louise Teresa Escobet, Xavier Olive. Diagnosability analysis based on component-supported analytical redundancy relations. Systems, Man and Cybernetics, Part A: Systems and Humans, IEEE Transactions. 2006;36(6):1146-1160.

Peer-review history:
The peer review history for this paper can be accessed here (Please copy paste the total link in your browser address bar)
http://sciencedomain.org/review-history/9921