The CSI Journal on
Computer Science and Engineering
Vol. 13, No. 2, 2016
Pages 23-31
Regular Paper

Taxonomy and Overview of Distributed Malfunction Diagnosis in Networks of Intelligent Nodes

Behrooz Parhami Nan Wu Sixin Tao

Department of Electrical and Computer Engineering, University of California, Santa Barbara, California, USA

Abstract

Started 50 years ago, the field of (system-level) malfunction diagnosis has expanded immensely and continues to be a very active subfield in both parallel processing and dependable computing research communities, with much of the new research coming from China and Taiwan in recent years. This paper represents an attempt to organize the field of research in distributed malfunction diagnosis via an overarching, descriptive, and consistent taxonomy that not only covers all of the past work, but also foretells of possible future research to fill gaps left by current results and areas that are just beyond the domains already investigated. The paper is accessible to computer science and engineering specialists who are new to the field, because it uses analogies to unveil the nature of the research problems and pertinent challenges.

Keywords: Comparison-Based Diagnosis, Diagnosability, Distributed System, Fault Tolerance, Interconnection Network, Malfunction Diagnosis, MM* Model, Parallel Processing, PMC Model, Self-Diagnosis, System-Level Fault Diagnosis.

1. Introduction

In the freshman seminar “Ten Puzzling Problems in Computer Engineering,” designed 10 years ago and taught since by the first author [23] [24], mathematical and logical puzzles are used to introduce advanced science/technology topics in a manner understandable to first-year college students. Two of the seminar’s puzzle types are relevant to the topic of this paper. One puzzle type places you on a remote island inhabited by members of two tribes, Truth-tellers and Liars. Truth-tellers provide the correct answer to any question, while Liars always give an untruthful answer. Members of the two tribes recognize each other, but you have no basis to judge which tribe a particular person belongs to, except by analyzing answers to questions you ask. A more advanced version of these puzzles postulates the tribes Truth-tellers and Randoms (they lie or tell the truth, completely at random). It turns out that Randoms are more difficult to deal with, because the consistency of Liars in providing untruthful answers is actually helpful in making deductions. The latter version of these puzzles models diagnosis in a distributed environment: You ask each node to perform self-diagnosis and report the result to you. A healthy node gives you a truthful answer about it being healthy, whereas a malfunctioning node gives you an untrustworthy answer. Is it possible to deduce which nodes are malfunctioning based on the responses received? The short answer is no, if there is no cross-checking of results.

Another puzzle asks you to imagine n people, mostly medical doctors (MDs), but mixed with a small number of impostors, sitting at a round table. Each person is told to interview the person seated to his/her right and render a judgment on whether that person is an MD or an impostor. Let’s assume that an MD knows how to question a person to determine with absolute certainty whether that person is indeed an MD. The n judgments are given to you and you must identify the impostors. Clearly, a judgment provided by an impostor is untrustworthy, much like answers provided by Randoms in the previous set of puzzles, not only because s/he does not have the knowledge to judge, but also because s/he may actually want to deceive you in order to remain undetected. This puzzle models the malfunction diagnosis problem as a directed graph G = (V, E), where vertices in V are intelligent nodes capable of testing each other and edges in E define a testing relation, with the directed edge (u, v) representing node u testing node v.

Five decades ago, this notion was formalized by Preparata, Metze, and Chien [27] into what has come to be known as the PMC model of malfunction diagnosis. Subsequently, Maeng and Malek [19] [20] devised a different formal model in which diagnosis is based on a managing unit comparing responses from two other units to which it is connected, concluding that the two responding units are healthy if their responses match and at least one of them malfunctioning otherwise. The model was subsequently refined and given the name MM* or comparison-based malfunction diagnosis model. As before, if the test manager is itself malfunctioning, no reliable conclusion can be reached.

An unfortunate side effect of the rapid advances in the field of distributed malfunction diagnosis is the emergence of a rather non-descriptive, and at times misleading, terminology. To cite one example, the two terms t/k-diagnosability and t/s-diagnosability mean different things, and the distinction of k versus s is lost when the parameters are replaced with actual numbers in a specific case; e.g., is 5/6 diagnosability of the first kind (t = 5, k = 6) or of the second kind (t = 5, s = 6)? Furthermore, the qualifiers "one-step," "sequential," and "pessimistic," applied to some kinds of diagnosis strategies discussed are rather undescriptive.

In this paper, we propose a taxonomy of malfunction diagnosis methods to facilitate understanding and contributing new results to the field. As a byproduct of the taxonomy, we expose certain areas of the field that need to be studied or explored in greater depth. This is not intended to be a complete survey of the field, as there have been literally hundreds of research contributions in the area of malfunction diagnosis over the past five decades. References cited are meant to cover pioneering contributions that have defined the field as a whole or its various sub domains, or have introduced new concepts, plus a few sources that support our contention that a new, descriptive nomenclature and taxonomy is indeed required.

2. What Is Malfunction Diagnosis?

Fault testing and fault diagnosis have been with us for centuries in connection with gadgets and systems whose designs are to be verified at the outset and whose correct functioning must be ascertained in the field as they are put to use. The term "fault" is a bit overused, as it has been applied at various levels of a digital system hierarchy, from devices and circuits to sizable modules incorporating hardware and software components. Fault testing in circuit and logic entail different methods than testing of higher-level modules. In fact, in modern practice, we often don't care about diagnosing a fault (identifying its location) within a circuit, say, a chip. Rather, we perform what is known as a go/no-go test that merely indicates whether the circuit is usable, replacing the entire circuit in case of a no-go result.

At the system level, by contrast, we do want to identify which module is causing problems, so that we can isolate and eventually repair/replace it. This requires a more elaborate diagnostic testing, instead of the go/no-go variety. For this reason, the term "system-level fault diagnosis" has been used for the latter situation. In the first author's nearly completed book on dependable computing [25], the term "malfunction diagnosis" is used to refer to the context above, avoiding the overuse of the term "fault" and obviating the need for the qualifier "system-level." So, our "malfunction diagnosis" is "system-level fault diagnosis" in much of the published literature. This use of malfunction diagnosis is the first element of our nomenclature and taxonomy.

Let us begin with the basic terminology and assumptions. We consider a system composed of interconnected, intelligent modules, where by intelligent we mean modules with internal processing and decision-making abilities. This isn't a restrictive assumption, as modern digital systems are composed of interconnection of processors, memory modules, I/O units, and the like, each having hardware control for basic functions and software control for functions that are not speed-critical and/or need flexibility over time. Each module is assumed to be capable of running a sophisticated self-test routine, when prompted, and to report the result to other modules.

3. Reflective vs. Comparative Models

Throughout our discussions, each test is assumed to return a yes/no value, indicating that all is good (yes = 0) or something is wrong (no = 1). If there are q tests, then the syndrome is a q-bit vector S with S[j] holding the result of test j. The diagnosis problem is to deduce from the binary syndrome vector S[1:q] the diagnosis vector D[1:n] reflecting the health (0) or non-health (1) of each of the n modules in the system.

In the reflective mode of diagnosis, known in the literature as the PMC model [27], when a module is connected to another module, we assume that one is capable of testing the other one. Actually, not all links may be usable as testing links and a sub graph of the directed graph representing the system may be designated as the testing graph. In fact, the connectivity of the system may be completely different from the testing graph. It is possible, for example, for the n nodes to be connected via a bus, so that each node can potentially test any other one. This situation can be represented by K_n, the n-node complete graph, assuming that the single bus cannot be a source of problems in testing; that is, it is modeled either as a malfunction-free system core or a set of n(n – 1) independent directed channels.

From now on, we focus on the testing graph only and ignore the fact that there may be other links in the system besides those used for testing or that the hardware connectivity may in fact be less dense than the testing graph. The nature of the test can vary, from significant interaction of passing back and forth test patterns and test outcomes to minimal interaction, with one module initiating the test (perhaps by sending a key or seed value) and the target module carrying out a self-test routine. The key or seed value serves to ensure that the test result isn't a constant that a malfunctioning module may produce by accident or from a previously stored result in memory, thus compromising diagnostic accuracy.

The abstract reflective testing relationship is shown in figure 1a, where details of how a test is performed are suppressed and only the yes/no or 0/1 conclusion from the test is deemed relevant. The comparative testing relationship can be abstracted as in figure 1b, where a test manager u and two participants v and w are involved. The node u initiates the testing and the nodes v and w respond to it by each sending a test result to u. If the two test results are identical, u concludes that all is well, producing the decision 0, provided u itself is not malfunctioning. Non-matching results lead to the 1 decision by u. If the manager u is malfunctioning, then we make no assumption about the decision it might produce [19] [20].

In the reflective model, the tests correspond to the edges of the testing graph, one test per edge. Thus, we have |E| = q, the number of tests. In comparative testing, however, triples (u, v, w) of nodes correspond to tests, with the triples used pre-defined as part of the diagnostic scheme. Viewed in this way, we immediately see that the 2-way and 3-way relationships of reflective and comparative testing can readily be generalized to higher-degree collaborative testing, where clusters of nodes perform intra-cluster testing according to some local schema and the overall result is deduced from the collection of cluster-level tests. If clusters constitute replaceable units within our system, then it does not matter which nodes within a cluster are malfunctioning. Those can be diagnosed off-line and the requisite repairs performed in parallel with a new replacement cluster taking over.

4. One-Step vs. Multi-Step Diagnosis

Conceptually, the simplest diagnostic scheme is when a single round of q tests are performed and the resulting binary syndrome vector S[1:q] is used to deduce which nodes are healthy and which are malfunctioning. When the information in the syndrome vector is always enough to do the required diagnosis for up to t malfunctions, we say that the system is one-step t-diagnosable. Necessary and sufficient conditions are known for one-step diagnosability, that is, the mapping of the syndrome vector S[1:q] into the diagnostic vector D[1:n], which correctly identifies the health (0) or malfunctioning (1) status of each unit. Theorem 1 represents an example of theoretical results that are available for practical use.

Theorem 1. An n-unit system in which no two units test one another, is 1-step t-diagnosable if and only if each unit is tested by at least t other units.

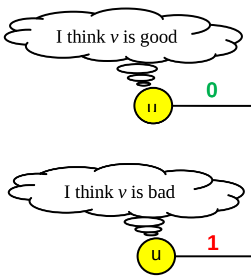
If, on the other hand, the syndrome vector isn't sufficient for full diagnosis but always leads to the identification of at least h malfunctioning units, $h < m$, we say that the system is multi-step t-diagnosable, because once the identified malfunctioning units have been repaired or replaced, the resulting system, which now has fewer malfunctioning units, can be subjected to the same process for identifying additional malfunctions. The extreme case where each diagnosis step identifies a single malfunctioning unit is referred to as "sequential diagnosis." A system is sequentially diagnosable if there exist a diagnosis strategy for it that guarantees the identification of at least one malfunctioning unit in each diagnosis step. Theorem 2 represents an example of theoretical results that are available with regard to sequential diagnosis, in this case a sufficient condition for sequential t-diagnosability.

Theorem 2. An n-unit system is sequentially t-diagnosable if the condition $n \geq 2t + 1$ holds. A majority of the n units being healthy is a sufficient condition, but it may not be necessary.

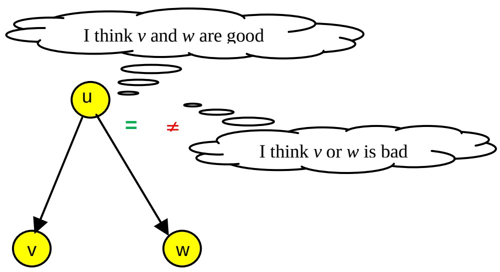
5. Sensitivity vs. Specificity of Diagnosis

The terms "sensitivity" and "specificity" are taken from the medical diagnosis domain. Suppose we have a population of individuals, mostly healthy but containing some who are afflicted with a particular disease. A medical test exists for the disease. The test can identify people afflicted with the disease (positive indication, or 1) and those not afflicted (negative indication, or 0), but it has some probability of yielding a false positive (identifying a healthy person as sick) and a certain probability of yielding a false negative (missing the detection of a sick person). Such a test is referred to as "sensitive" if it has a fairly small false-negative probability, that is, it detects nearly all sick individuals (Figure 2a). The test is dubbed "specific" if it has a fairly small false-positive possibility, that is, only a minute fraction of healthy individuals will be wrongly diagnosed as having the disease (Figure 2b).

In the context of studies on malfunction diagnosis, false negatives have not been allowed so far. Put another way, the diagnosis outcome can have healthy nodes marked as bad (this is a safe situation) but no malfunctioning node is allowed to be misidentified as healthy. However, there is no fundamental reason for excluding false negatives, if the system has some built-in malfunction tolerance capability that allows it to function correctly in the presence of a very small number of malfunctioning units. Such a system will use a combination of malfunction masking and malfunction diagnosis to continue correct operation in the presence of some malfunctions, aiming to remove malfunctions that put it over its tolerance capacity.



(a) Reflective testing (PMC model)



(b) Comparative testing (MM* model)

Figure 1. Reflective and comparative testing abstractions

(a) A highly sensitive test (b) A highly specific test

Figure 2. Diagnostic sensitivity and specificity

6. Unrestricted vs. Conditional Malfunction Patterns

If the subset of m malfunctioning units can be arbitrary, the diagnosis scheme is unrestricted. This is the default assumption for any diagnosis scheme in which no restriction is mentioned.

The main kind of conditional diagnosis schemes studied thus far is when the m malfunctions are restricted not to include all neighbors of any node. When all neighbors of a node are malfunctioning, that node becomes isolated from healthy units and thus cannot be correctly diagnosed. This isolation poses a problem for the diagnosis algorithms, effectively restricting t to at most $d - 1$, where d is the minimum node degree, when no false positives are allowed. Recently, a stronger restriction, requiring each node to have at least g good neighbors, has been proposed. The g-good-neighbor diagnosability schemes requires each node to have at least g good neighbors, in which case t-diagnosability for larger values of t can be ensured. The previously-studied "conditional" diagnosability corresponds to the special 1-good-neighbor case of this more general scheme.

Unrestricted and conditional diagnosabilities can be combined in many different ways. For example, it is possible to prove that certain classes of networks are $(t + a)$-diagnosable, except when the pattern of malfunctions belongs to some undesirable class, in which case they become t-diagnosable. In other words, the absence of the undesirable malfunction patterns increases the diagnosability extent by a. An example of such combining is "strong diagnosability," where the level of diagnosability rises from t to $t + 1$ (that is, $a = 1$ in the formulation above) when every node possesses at least one healthy neighbor.

Again, more general conditions can be entertained. In a cluster-based hierarchical network, one may postulate that not all nodes in any given cluster be malfunctioning, that each cluster remain connected internally, or that at least one inter-cluster connection remain intact between any two clusters. The possibilities are quite varied. In general, a restriction on the malfunction pattern leads to some increase in the diagnosability extent.

7. Analysis vs. Synthesis Considerations

Diagnosability problems to be solved are of two types: analyzing diagnosabilities of known networks, and synthesizing interconnection architectures with desired diagnosability properties.

Analysis problem 1: Given a syndrome vector S[1:q], identify a set M that includes the requisite number of malfunctioning nodes (m, 1, or some other number, depending on the model used and the diagnostic strategy). Note that the suspected malfunction set M may be allowed to include false positives or prohibited from signaling false negatives.

In the simplest case, polynomial-time algorithms exist that take the vector S[1:q] and the testing graph as input and produce the set M when the set is restricted to contain all and only the m malfunctioning units. Efficient algorithms exist for certain other cases as well, though the space of possibilities has not been exhausted at this writing.

Analysis problem 2: Given a testing structure (testing graph of PMC, 3-groupings for MM*), identify the extent of diagnosability in the case of one-step, multi-step (including sequential), and other strategies for various unrestricted and conditional patterns of malfunctions.

Much work has been done in this area, including the derivation of diagnosability results for a wide array of known and newly proposed interconnection networks. The networks studied include meshes, tori [1], hypercubes [3] [11] [12] [14] [26] (or its generalizations [34] [36] [38]), k-ary n-cubes [1], numerous hypercube variants [16], cube-connected cycles, OTIS or swapped networks (including the biswapped variant), Cartesian product networks [1], and many other regular [5] [6] [18] [35] and hierarchical (multi-level) networks.

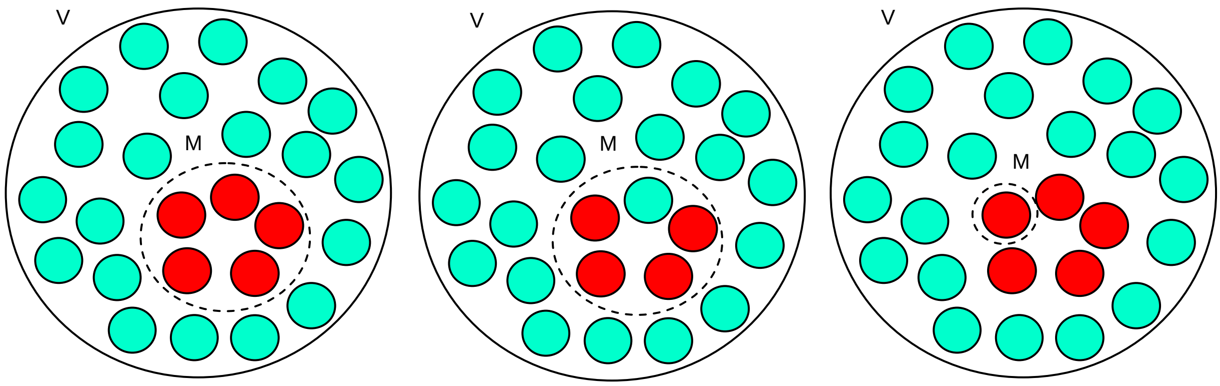
Synthesis problem: Given a desired diagnosability extent, the number of nodes, and other physical attributes, derive a testing graph that is optimal in some respect.

The synthesis problem is easy when only diagnosability is of interest, but becomes very challenging (like most combinatorial optimization problems) when other criteria are included.

8. How the Taxonomy Is Used

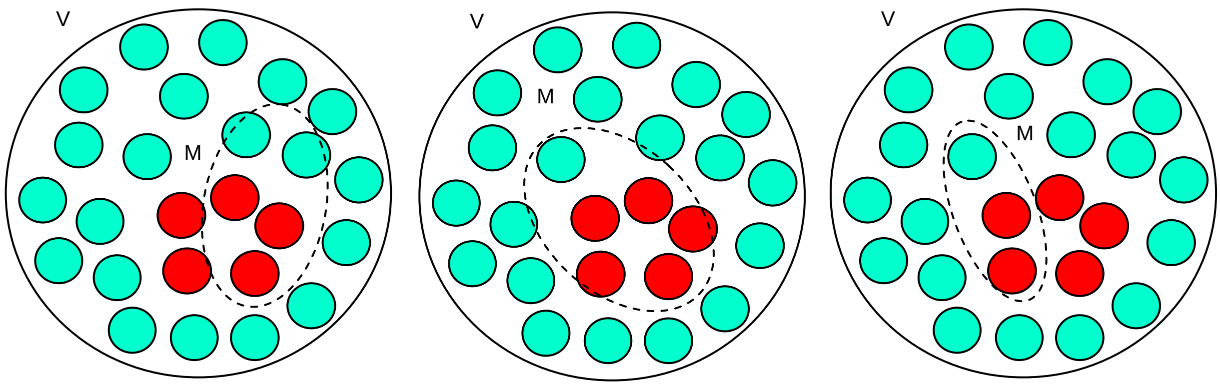
Our taxonomy essentially entails the mentioning of each of the four parameters t, T_p, F_p, and F_n in the form of $t/T_p/F_p/F_n$-diagnosability. These parameters also contain information about whether the scheme is one-step or multi-step (including sequential) and whether it is precise or pessimistic. This method of specifying a diagnostic scheme, including incorporation of the maximum number of false negatives as the last of four parameters is new. Existing diagnosis schemes do not allow false negatives (the corresponding number is 0 in our model), but, as mentioned in Section 5, there is no fundamental reason to exclude them forever. In the examples that follow, $F_n = 0$ and is thus not discussed explicitly.

The existing models correspond to the following scheme with our terminology:



(a) Precise 1-step 5-diagnosability (b) Pessimistic 1-step 5-diagnosability, |M| = t (c) Sequential 5-diagnosability

Figure 3. Some commonly studied diagnosis strategies and outcomes



(a) Imprecise 1-step 5-diagnosability (b) Pessimistic 5-diagnosability, |M| > t (c) Multi-step 5-diagnosability

Figure 4. Examples of diagnosis strategies allowing false negatives and |M| possibly going beyond t.

Precise = No false positives allowed, that is, $F_p = 0$

Pessimistic = Up to t – m or s – m (with s > t) false positives allowed

Example 1 (5/5/0/0-diangnosability): Up to 5 malfunctions are diagnosed with no false positives. This essentially specifies precise one-step 5-diagnosability with existing terminology (see figure 3a).

Example 2 (5/5/1/0-diagnosability): Up to 5 malfunctions are diagnosed, with the malfunctioning units identified to within a set of 5 units (up to 5 true positives and up to 1 false positive; see figure 3b).

Example 3 (5/1/0/0-disgnosability): One malfunction is diagnosed in each step, with no false positives. This corresponds to sequential 5-diagnosability (Figure 3c).

Example 4 (5/2/0/0-diagnosability): Up to 5 malfunctions are diagnosed, with the bad units identified in 3 steps (at least 2 true positives and no false negative at each step).

And here are a few examples, not yet studied, that entail false negatives.

Example 5 (5/5/2/2-diagnosability): Up to 5 malfunctions are allowed, with three of the malfunctioning units identified in one step to within a set of 5 units (at least 3 true positives and up to 2 false negatives; see figure 4a).

Example 6 (5/6/1/0-diagnosability): Up to 5 malfunctions are diagnosed, with the malfunctioning units identified to within a set of 6 units (up to one false positive; see figure 4b).

Example 7 (5/2/1/0-diagnosability): Up to 5 malfunctions are diagnosed in 3 steps, each step identifying 2 true positives and up to 1 false positive (Figure 4c).

9. Partial Survey of Prior Work

The references at the end of our paper contain a representative sample of work in the field of distributed malfunction diagnosis, both early work laying the foundations and more recent work developed within a mature field. It would be instructive to categorize these references with regard to our taxonomy. Tables 1 and 2 show the results of classification for reflective (PMC) and comparative (MM*) models of malfunction diagnosis.

Several patterns emerge from the survey of representative work reflected in tables 1 and 2. First, the synthesis problem has not received much attention, particularly within the comparative diagnosis model. Second, multi-step diagnosis, which is often a more difficult problem from a theoretical standpoint, has not been the focus of much work. Studies on single-step diagnosis are dominant, particularly with comparative methods. High-specificity diagnosis has received more attention than low-specificity versions.

It is also evident from tables 1 and 2 that the sensitivity of diagnosis has been completely ignored (this is why our tables do not include columns for this attribute).

Table 1. Categorization of prior work on reflective malfunction diagnosis (PMC model)

Reference	**Paper's Aim**	**Steps**	**Specificity**	**Qualification**
Citation	Analysis / Synthesis	Single / Multiple	High / Low	Unrestricted / Conditional
[1] Araki & Shibata 2000	Analysis	Single	Both	Unrestricted
[2] Araki & Shibata 2003	Analysis	Multiple	High	Unrestricted
[3] Armstrong & Gray 1981	Analysis	Single	High	Unrestricted
[4] Barsi et al. 1976	Synthesis	Both	High	Unrestricted
[5] G. Y. Chang et al. 2005	Analysis	Single	Both	Unrestricted
[6] G. Y. Chang 2010	Analysis	Multiple	High	Unrestricted
[7] N. W. Chang & Hsieh 2012	Analysis	Single	High	Conditional
[8] Hakimi & Amin 1974	Analysis	Single	High	Unrestricted
[13] Karunathini & Friedman 1979	Analysis	Both	Low	Unrestricted
[14] Kavianpour & Kim 1991	Analysis	Single	Low	Unrestricted
[15] Lai et al. 2005	Analysis	Single	High	Conditional
[16] Lin et al. 2014	Analysis	Single	High	Conditional
[17] Lin et al. 2015	Analysis	Single	High	Conditional
[18] Lin et al. 2016	Analysis	Single	Low	Unrestricted
[26] Peng et al. 2012	Analysis	Single	High	Conditional
[27] Preparata et al. 1967	Analysis	Both	High	Unrestricted
[30] Somani et al. 1987	Synthesis	Single	High	Unrestricted
[31] Somani et al. 1996	Analysis	Single	Low	Unrestricted
[32] Tsai & Chen 2013	Analysis	Single	Both	Unrestricted
[34] M. Xu et al. 2009	Analysis	Single	High	Conditional
[35] L. Xu et al 2016	Analysis	Single	Both	Both
[38] Zhu 2008	Analysis	Single	High	Conditional
[39] Zhu et al. 2014	Analysis	Single	High	Both

Table 2. Categorization of prior work on comparative malfunction diagnosis (MM* model)

Reference	**Paper's Aim**	**Steps**	**Specificity**	**Qualification**
Citation	Analysis / Synthesis	Single / Multiple	High / Low	Unrestricted / Conditional
[5] G. Y. Chang et al. 2005	Analysis	Single	Both	Unrestricted
[10] Hong & Hsieh 2012	Analysis	Single	High	Both
[11] Hsieh & Kao 2013	Analysis	Single	High	Conditional
[12] Hsu et al. 2009	Analysis	Single	High	Conditional
[17] Lin et al. 2015	Analysis	Single	High	Conditional
[19] Maeng & Malek 1981	Analysis	Single	High	Unrestricted
[20] Malek 1980	Analysis	Single	High	Unrestricted
[29] Sengupta & Dabbura 1992	Analysis	Single	High	Unrestricted
[36] Yang 2013	Analysis	Single	High	Conditional
[39] Zhu et al. 2014	Analysis	Single	High	Both
[40] Ziwich & Duarte 2016	Analysis	Single	High	Unrestricted

Other areas where there is no work yet include hierarchical or cluster-based diagnosis. Numerous hierarchical or multi-level interconnection schemes have been proposed based on hypercube [9] and its many variants [21] [28]. There are also interesting hierarchical interconnection architectures that are grown from arbitrary basis topologies. A prime example is the class of swapped or OTIS networks [22] [37], and their symmetric variants known as biswapped networks [33], which have been the subjects of very limited diagnosability studies [32].

10. Conclusion and Future Work

The nomenclature and taxonomy introduced in this paper puts the field of malfunction diagnosis into a much-needed order, allowing a uniform formulation of the problems already explored and the exposure of additional possibilities not yet investigated. The various diagnostic strategies are expressed in terms of the four parameters t, T_p, F_p, and F_n that collectively specify not only the extent of diagnosability but also whether the scheme is 1-step, multi-step, precise, or pessimistic in the prevailing terminology.

The idea of allowing false positives in the diagnostic scheme isn’t new, but the explication of the number of false positives allowed as a model parameter is helpful and removes some of the ambiguities in the current nomenclature. False positives aren’t as undesirable as they once were, given that the steep reduction in hardware cost makes system down time considerations much more important than the loss of a healthy unit. In fact, a unit falsely identified as malfunctioning may only be lost temporarily, because off-line testing can verify that the unit is in fact good, allowing it to return to the spare supply. False negatives, on the other hand are new to our model. The presence of some malfunctioning units may be tolerated by a system’s built-in malfunction tolerance, which may include replicated computation with voting or data replication with primary and back-up nodes.

We plan to work on further refining this taxonomy as we discover diagnostic schemes that it does not properly cover or see the need for additional expressive power as system complexity and diagnostic strategies evolve.

Appendix

List of Symbols

a Additional diagnosability beyond t under special circumstances
D Diagnosis binary vector of length n
d Minimum node degree in G
E Set of edges of the testing graph, with $|E| \geq q$
F_n Number of false negatives allowed by the testing strategy
F_p Number of false positives allowed by the testing strategy
G The testing directed graph, G = (V, E)
g Minimum number of good neighbors for each node assumed in some conditional models
h Minimum number of malfunctioning units (true positives) included in M
K_n Complete graph of n nodes
k Bound on the number of false positives in previous terminology (our F_p)
M Set of purportedly malfunctioning units returned by the diagnosis algorithm; $|M| = T_p + F_p$
m Actual number of malfunctioning units, $m \leq t$
n Number of nodes in the network or testing graph (length of the binary diagnosis vector D[1:n])
q Number of tests performed in one step (length of the binary syndrome vector S[1:q])
S Syndrome binary vector
s Bound on the size of the returned set M, with s > t
t Upper bound on the number of malfunctioning nodes
T_p Number of true positives (correctly diagnosed malfunctioning units) by the testing strategy
u Graph node doing the testing or coordination
V Set of system nodes, with |V| = n
v Graph node under test by u
w Second graph node under test by u in the comparative model

References

[1] T. Araki, and Y. Shibata, "Diagnosability of Networks Represented by the Cartesian Product," *IEICE Trans. Fundamentals*, vol. E83-A, no. 3, pp. 465-470, 2000.

[2] T. Araki, and Y. Shibata, "(*t*, *k*)-Diagnosable System: A Generalization of the PMC Models," *IEEE Trans. Computers*, vol. 52, no. 7, pp. 971-975, 2003.

[3] J. R. Armstrong, and F. G. Gray, "Fault Diagnosis in a Boolean *n*-Cube Array of Microprocessors," *IEEE Trans. Computers*, vol. 30, no. 8, pp. 587-590, 1981.

[4] F. Barsi, N. Grandoni, and P. Maestrini, "A Theory of Diagnosability of Digital Systems," *IEEE Trans. Computers*, vol. 25, no. 6, pp. 585-593, 1976.

[5] G.-Y. Chang, G. J. Chang, and G.-H. Chen, "Diagnosabilities of Regular Networks," *IEEE Trans. Parallel and Distributed Systems*, vol. 16, no. 4, pp. 314-323, 2005.

[6] G. Y. Chang, "(*t*, *k*)-Diagnosability for Regular Networks," *IEEE Trans. Computers*, vol. 59, no. 9, pp. 1153-1157, 2010.

[7] N. W. Chang, and S. Y. Hsieh, "Conditional Diagnosability of Augmented Cubes under the PMC Model," *IEEE Trans. Dependable and Secure Computing*, vol. 9, no. 1, pp. 46-60, 2012.

[8] S. L. Hakimi, and A. T. Amin, "Characterization of Connection Assignment of Diagnosable Systems," *IEEE Trans. Computers*, vol. 23, no. 1, pp. 86-88, 1974.

[9] J. P. Hayes, and T. Mudge, "Hypercube Supercomputers," *Proc. IEEE*, vol. 77, no. 12, pp. 1829-1841, 1989.

[10] W. S. Hong, and S. Y. Hsieh, "Strong Diagnosability and Conditional Diagnosability of Augmented Cubes under the Comparison Diagnosis Model," *IEEE Trans. Reliability*, vol. 61, no. 1, pp. 140-148, 2012.

[11] S. Y. Hsieh, and C. Y. Kao, "The Conditional Diagnosability of *k*-ary *n*-cubes under the Comparison Diagnosis Model," *IEEE Trans. Computers*, vol. 62, no. 4, pp. 839-843, 2013.

[12] G. H. Hsu, C. F. Chiang, L. M. Shih, L. H. Hsu, and J. J. Tan, "Conditional Diagnosability of Hypercubes under the Comparison Diagnosis Model," *J. Systems Architecture*, vol. 55, no. 2, pp. 140-146, 2009.

[13] S. Karunanithi, and A. D. Friedman, "Analysis of Digital Systems Using a New Measure of System Diagnosis," *IEEE Trans. Computers*, vol. 28, no. 2, pp. 121-133, 1979.

[14] A. Kavianpour, and K. H. Kim, "Diagnosabilities of Hypercubes under the Pessimistic One-Step Diagnosis Strategy" *IEEE Trans. Computers*, vol. 40, no. 2, pp. 232-237, 1991.

[15] P. L. Lai, J. J. Tan, C. P. Chang, and L. H. Hsu, "Conditional Diagnosability Measures for Large Multiprocessor System," *IEEE Trans. Computers*, vol. 54, no. 2, pp. 165-175, 2005.

[16] L. Lin, S. Zhou, L. Xu, and D. Wang, "Conditional Diagnosability of Arrangement Graphs under the PMC Model," *Theoretical Computer Science*, vol. 548, pp. 79-97, 2014.

[17] L. Lin, L. Xu, D. Wang, and S. Zhou, "The *g*-Good-Neighbor Conditional Diagnosability of Arrangement Graphs," *IEEE Trans. Dependable and Secure Computing*, December 2015.

[18] L. Lin, L. Xu, S. Zhou, and S. Y. Hsieh, "The *t*/*k*-Diagnosability for Regular Networks," *IEEE Trans. Computers*, vol. 65, no. 10, pp. 3157-3170, 2016.

[19] J. Maeng, and M. Malek, "A Comparison Connection Assignment for Self-Diagnosis of Multiprocessor Systems," *Proc. 11th Int'l Symp. Fault-Tolerant Computing*, 1981, pp. 173-175.

[20] M. Malek, "A Comparison Connection Assignment for Diagnosis of Multiprocessor Systems," *Proc. 7th Symp. Computer Architecture*, pp. 31-36, 1980.

[21] B. Parhami, *Introduction to Parallel Processing: Algorithms and Architectures*, Plenum, 1999.

[22] B. Parhami, "Swapped Interconnection Networks: Topological, Performance, and Robustness Attributes," *J. Parallel and Distributed Computing*, vol. 65, no. 11, pp. 1443-1452, 2005.

[23] B. Parhami, "A Puzzle-Based Seminar for Computer Engineering Freshmen," *Computer Science Education*, vol. 18, no. 4, pp. 261-277, 2008.

[24] B. Parhami, "Motivating Computer Engineering Freshmen Through Mathematical and Logical Puzzles," *IEEE Trans. Education*, vol. 52, no. 3, pp. 360-364, 2009.

[25] B. Parhami, *Dependable Computing: A Multi-Level Approach*, graduate-level textbook in development, available on-line at: http://www.ece.ucsb.edu/~parhami/text_dep_comp.htm.

[26] S.-L. Peng, C.-K. Lin, J. J. M. Tan, and L.-H. Hsu, "The g-Good-Neighbor Conditional Diagnosability of Hypercube under PMC Model," *Applied Mathematics and Computation*, vol. 218, pp. 10406-10412, 2012.

[27] F. P. Preparata, G. Metze, and R. T. Chien, R. T., "On the Connection Assignment Problem of Diagnosable Systems," *IEEE Trans. Electronic Computers*, vol. 16, no. 6, pp. 848-854, 1967.

[28] F. P. Preparata, and J. Vuillemin, "The Cube-Connected Cycles: A Versatile Network for Parallel Computation," *Communications of the ACM*, vol. 24, no. 5, pp. 300-309, 1981.

[29] A. Sengupta, and A. T. Dahbura, "On Self-Diagnosable Multiprocessor Systems: Diagnosis by the Comparison Approach," *IEEE Trans. Computers*, vol. 41, no. 11, pp. 1386-1396, 1992.

[30] A. K. Somani, V. K. Agarwal, and D. Avis, "A Generalized Theory for System Level Diagnosis," *IEEE Trans. Computers*, vol. 36, pp. 538-546, 1987.

[31] A. K. Somani, and O. Peleg, "On Diagnosability of Large Fault Sets in Regular Topology-Based Computer Systems," *IEEE Trans. Computers*, vol. 45, no. 8, pp. 892-903, 1996.

[32] C.-H. Tsai, and J.-C. Chen, "Fault Isolation and Identification in General Biswapped Networks under the PMC Diagnostic Model," *Theoretical Computer Science*, vol. 501, pp. 62-71, 2013.

[33] W. J. Xiao, B. Parhami, W. D. Chen, M. X. He, and W. H. Wei "Fully Symmetric Swapped Networks Based on Bipartite Cluster Connectivity," *Information Processing Letters*, vol. 110, no. 6, pp. 211-215, 2010.

[34] M. Xu, K. Thulasiraman, and X. D. Hu, "Conditional Diagnosability of Matching Composition Networks under the PMC Model," *IEEE Trans. Circuits and Systems II*, vol. 56, no. 11, pp. 875-879, 2009.

[35] L. Xu, L. Lin, S. Zhu, and S.-Y. Hsieh, "The Extra Connectivity, Extra Conditional Diagnosability, and *t*/*m*-Diagnosability of Arrangement Graphs," *IEEE Trans. Reliability*, vol. 65, no. 3, pp. 1248-1262, September 2016.

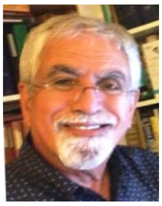
[36] M. C. Yang, "Conditional Diagnosability of Matching Composition Networks under the MM∗ Model," *Information Sciences*, vol. 233, pp. 230-243, 2013.

[37] F. Zane, P. Marchand, R. Paturi, and S. Esener, "Scalable Network Architectures Using the Optical Transpose Interconnection System (OTIS)," *J. Parallel and Distributed Computing*, vol. 60, no. 5, pp. 521-538, 2000.

[38] Q. Zhu, "On Conditional Diagnosability and Reliability of the BC Networks," *J. Supercomputing*, vol. 45, no. 2, pp. 173-184, 2008.

[39] Q. Zhu, G. Guo, and D. Wang, "Relating Diagnosability, Strong Diagnosability and Conditional Diagnosability of Strong Networks," *IEEE Trans. Computers*, vol. 63, no. 7, pp. 1847-1851, 2014.

[40] R. P. Ziwich, and E. P. Duarte, "A Nearly Optimal Comparison-Based Diagnosis Algorithm for Systems of Arbitrary Topology," *IEEE Trans. Parallel and Distributed Systems*, vol. 27, no. 11, pp. 3131-3143, November 2016.



Behrooz Parhami (PhD, UCLA 1973) is Professor of Electrical and Computer Engineering, and former Associate Dean for Academic Personnel, College of Engineering, at University of California, Santa Barbara, where he teaches and does research in computer arithmetic, parallel processing, and dependable computing. A Life Fellow of IEEE, a Fellow of IET and British Computer Society, and recipient of several other awards (including a most-cited paper award from J. Parallel & Distributed Computing), he has written six textbooks and more than 290 peer-reviewed technical papers. Professionally, he serves on journal editorial boards and conference program committees and is also active in technical consulting.

E-mail: parhami@ece.ucsb.edu

Nan Wu is a doctoral student in Computer Engineering at Department of Electrical and Computer Engineering, University of California, Santa Barbara. Born in 1994, she received a BE degree in electronics engineering from Tsinghua University, Beijing, China, in 2016. Her research interests include applications of emerging technologies and spiking neural networks.

E-mail: nanwu@umail.ucsb.edu

Sixin Tao is a master's student in Computer Engineering at Department of Electrical and Computer Engineering, University of California, Santa Barbara. Born in 1992 in Jiangsu Province, she received a Bachelor of Engineering degree from College of Automation Engineering, Nanjing University of Aeronautics and Astronautics, China, in 2015.

E-mail: sixin@umail.ucsb.edu

Paper Handling Data:

Submitted: 28.11.2016
Received in revised form: 16.12.2016
Accepted: 02.01.2017
Corresponding author: Prof. Behrooz Parhami, Department of Electrical and Computer Engineering, University of California, Santa Barbara, California, USA.