Parallel Processing Letters Vol. 3 No. 2 (1993) 195–204

DISTRIBUTED FAULT DIAGNOSIS OF A RING OF PROCESSORS

A. DAS
DIRO, University of Montreal
Montreal, H3C 3J7, Canada

K. THULASIRAMAN
Department of Electrical Engineering, Concordia University
Montreal, H3G 1M8, Canada

K. B. LAKSHMANAN
Department of Mathematics and Computer Science, SUNY at Brockport
Brockport, New York 14420, USA

and

V. K. AGARWAL
Department of Electrical Engineering, McGill University
Montreal, H3A 2A7, Canada

Received 2 June 1992
Revised 23 March 1993
Accepted by S. Das

ABSTRACT

Based on the theory of system level diagnosis pioneered by Preparata, Metze and Chien [1], we study fault diagnosis of a ring of processors. We use the comparison based model of Chwa and Hakimi [4]. It is shown that the processors in a ring can be diagnosed uniquely if, and only if, the faults are distributed in a certain manner. We present algorithms to diagnose the faulty processors when these requirements are satisfied. These algorithms permit distributed implementation on both unidirectional and bidirectional rings.

1. Introduction

Continuing advances in semi-conductor technology have now made available large multiprocessor systems such as the hypercube systems. The increasing complexity of these systems poses challenging problems in ensuring their reliability. The problems of fault detection, diagnosis and reconfiguration of multiprocessor systems have thus become active areas of intensive research in recent years. Various models of fault diagnosis have been studied and significant algorithms and related complexity results have also been reported [1–6]. The pioneering work of Preparata, Metze and Chien [1] continues to be the guiding force in all these studies.

In multiprocessor systems, such as those implementable in very large scale integration (VLSI) and wafer-scale integration (WSI), the number of processors in a system can be very large. Moreover, the commonly used system interconnection networks such as the rectangular grids are very symmetrical and sparse. When such a system is analyzed using the classical theory, the number of faulty processors permitted is very small in comparison to the number of processors in the system. This shortcoming motivated the recent works on probabilistic diagnosis algorithms for sparsely interconnected systems [7,8].

Most diagnosis algorithms are assumed to be executed on a single highly reliable supervisory processor. A single supervisory processor is a performance bottleneck in systems with a large number of processing elements. Distributed diagnosis algorithms executed on the multiprocessor itself would be desirable. Works motivated by this consideration may be found in [9,10].

Our work in [11] is also motivated by the inadequacy of the classical approach when applied to large sparsely interconnected systems, as well as the need for distributed diagnosis algorithms. In this work, we presented a theory of local diagnosis and introduced a class of locally diagnosable systems called t-in-L_1 diagnosable systems. A system is t-in-L_1 diagnosable if all the processors can be uniquely diagnosed as faulty or fault-free provided that there are at most t faulty processors in the local domain $L_1(u) \cup \{u\}$ of each processor u, where $L_1(u)$ is the set of processors adjacent to u. The main result of [11] is that regular systems such as the hypercube, the rectangular, hexagonal and orthogonal grids in t-in-L_1 diagnosable for $t = \lfloor |L_1(u)|/2 \rfloor + 1$ for each processor u provided the number of faulty processors is less than $n/2$, where n is the total number of processors in the system. Sufficient conditions for a system to be t-inL_1 diagnosable are also presented in [11]. The local diagnosis algorithms of [11] can be executed in a distributed manner in the multiprocessor itself. However, the results presented in [11] require that each processor be adjacent to at least three processors, and this is not applicable when the processors form a ring.

In this paper, we discuss diagnosis of a ring of processors under local fault constraints. Our diagnosis algorithms permit a distributed implementation. We use the comparison based model of Chwa and Hakimi [4].

The paper is organised as follows. In Sec. 2, the basic model and definitions are presented. In Sec. 3, we discuss a local diagnosis algorithm. We present in Sec. 4 the diagnosis of a ring of processors.

2. Preliminaries

A multiprocessor system S consists of n independent processors $U = \{u_1, u_2, \ldots, u_n\}$. In the comparison model of multiprocessor fault diagnosis [4], all processors in S are assigned to perform the same task. Upon completion, the outputs of neighboring pairs of these processors are compared. The collection of these outputs is called a syndrome. The comparison assignment can be represented by an undirected graph $G = (U, E)$ where an edge e_{ij} belongs to E if, and only if, the outputs of u_i and u_j are compared. The distance in G between processors u_i and u_j refers to the minimum number of edges in any path between u_i and u_j. The distance between two processors u_i and u_j is denoted by $d(u_i, u_j)$. A processor $u_i \neq u_j$ is said to belong to a local domain $L_k(u_j)$ if u_i lies within a distance k of u_j. An outcome a_{ij} is associated with each pair of processors whose outputs are compared, where $a_{ij} = 0(1)$ if the outputs compared agree(disagree). Only permanent faults are considered and as in [5] we assume that the outputs of a fault-free and a faulty processor always disagree. It follows that $a_{ij} = 0$ whenever both u_i and u_j are fault-free; $a_{ij} = 1$ if one of u_i and u_j is fault-free and the other faulty; a_{ij} is unreliable if both u_i and u_j are faulty. An edge that has a 0(1) outcome associated with it is referred to as a 0-link(1-link). $N_0(u_i)$ and $N_1(u_i)$ denote the sets of processors adjacent to u_i and connected to u_i by a 0-link and a 1-link respectively. For $X \subseteq U$, $N_0(X) = \{u_j | u_j \in N_0(u_i)$ for some $u_i \in X\}$, and $N_1(X) = \{u_j | u_j \in N_1(u_i)$ for some $u_i \in X\}$.

Paths starting from processor u_i are said to be **distinct** if, and only if, they have no vertex in common other than u_i. A fault set $F \subseteq U$ is a **permissible fault set** for a set of fault constraints if F satisfies the requirements of the fault constraints. Given a syndrome, F is an **allowable fault** set if, and only if, F is a permissible fault set, and the assumption that the processors in F are faulty and the processors in $U-F$ are fault-free is consistent with the given syndrome. The complement of a set F will be denoted by F^c.

We use the following notation for the figures: a dot within a circle represents a fault-free processor and an x within a circle represents a faulty processors.

3. A Local Diagnosis Algorithm

Let S be a multiprocessor system with test interconnection graph $G = (U, E)$. Note that when the comparison model is used, G is an undirected graph.

Let u_i be a processor in S with t distinct paths of length 2 from $u_i \cdot R_t(u_i)$, called a **neighborhood of order t** around u_i, denotes the set of processors which lie on these paths including u_j. Intuitively, we can determine the faulty or fault-free status of u_i using only the comparison outcomes among processors in $R_t(u_i)$ as follows. Consider the processors on a distinct path of length 2 from u_i. Assuming u_i to be fault-free, we can try to identify the faulty processors among the two other processors on the path. If u_i is actually fault-free and the path contains faulty processors, then at least one can be identified as faulty. If u_i is actually faulty, then exactly one processor corresponding to each fault-free processor can be identified as faulty. Each distinct path is treated similarly. Thus, if more than half the processors in $R_t(u_i)$ are fault-free, then the status of u_i can be determined; u_i is fault-free if, and only if, at most half the processors in $R_t(u_i)$ are identified as faulty.

Theorem 1. Let u_i be a processor in a system S with t distinct paths of length 2 from u_i. Let $R_t(u_i)$ denote the set of processors which lie on these paths including processor u_i. If at most t processors are faulty in $R_t(u_i)$, then u_i is faulty if, and only if,

$$\left|F(u_i)\bigcap R_t(u_i)\right| > t$$

where $F(u_i)$ is the set of processors which have a 1-link with u_i or can be reached from u_i by a 1-link followed by a 0-link or by a 0-link followed by a 1-link.

Proof. Recall that $N_0(u_i)$ and $N_1(u_i)$ denote the sets of processors which are incident to the processor u_i by a 0-link and a 1-link respectively. Then $F(u_i) = N_1(u_i) \cup N_1(N_0(u_i)) \cup N_0(N_1(u_i))$. We observe that if u_i is fault-free, then $F(u_i) \cap R_t(u_i)$ denotes the set of processors which can immediately be declared as faulty. Hence, if $|F(u_i) \cap R_t(u_i)| > t$, then u_i is faulty, for otherwise it contradicts the fact that at most t processors can be faulty in $R_t(u_i)$. □

Now, suppose we assume u_i to be faulty. Consider a pair (u_{j1}, u_{j2}) in $R_t(u_i)$, with u_{j1} adjacent to u_i.

Case 1. u_{j1} is fault-free and u_{j2} is faulty: Clearly u_{j1} is in $N_1(u_i)$ and hence belongs to $F(u_i) \cap R_t(u_i)$.

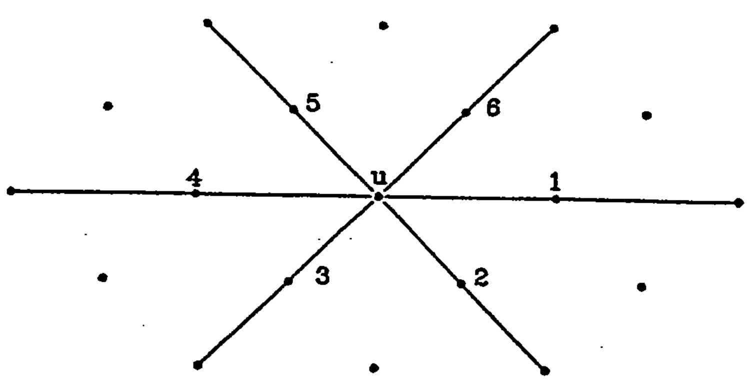
Case 2. u_{j1} is fault-free and u_{j2} is fault-free: Clearly u_{j1} is in $N_1(u_i)$ and u_{j2} belongs to $N_0(N_1(u_i))$. Thus, both u_{j1} and u_{j2} belong to $F(u_i) \cap R_t(u_i)$.

Case 3. u_{j1} is faulty and u_{j2} is fault-free: In this case, since both u_i and u_{j1} are faulty, u_{j1} may belong to either $N_0(u_i)$ or $N_1(u_i)$. If u_{j1} is in $N_0(u_i)$, then u_{j2} is in $N_1(N_0(u_i))$, and hence also belongs to $F(u_i) \cap R_t(u_i)$. On the other hand, if u_{j1} is in $N_1(u_i)$, it itself rather than u_{j2} belongs to $F(u_i) \cap R_t(u_i)$.

Thus, in all three cases above, we find that if u_i is faulty, then for every fault-free processor in $R_t(u_i)$, there exists a corresponding processor in $F(u_i) \cap R_t(u_i)$. Also since at most t processors can be faulty in $R_t(u_i)$, there are at least $t+1$ fault-free processors. As a result, if u_i is faulty, then $|F(u_i) \cap R_t(u_i)| > t$. This completes the proof. □

The diagnosis result presented above permits correct diagnosis of a processor, as long as a local neighborhood $R_t(u_i)$ of order t can be defined around u_i and it contains at most t faulty processors. Clearly, the value of t can be different for different processors. Moreover, the neighborhood can be defined in a variety of ways. However, for regular interconnected structures it is convenient to predefine a local neighborhood of the same order around each processor in a uniform way so that an algorithm that works in a distributed manner can be implemented. If the local neighborhood around each processor can be constructed to have the same topology, then each processor can execute a copy of the same local diagnosis algorithm synchronously. Initially, each processor must execute the same job and transmit the result to each of its neighbors so that the results can be compared and the comparison outcomes generated. At this point the comparison outcomes are available in a distributed manner with each bit of the comparison outcome being available at two sites, namely the processors involved in that comparison. In other words, each processor has t bits of comparison syndrome, corresponding to the comparison tests with its t neighbors. These are used to compute $N_0(u_i)$ and $N_1(u_i)$. In order to compute $N_0(N_1(u_i))$ and $N_1(N_0(u_i))$, each processor must receive some information from each of its neighbors. But depending on the the pairing of the processors in $R_t(u_i)$, only one bit of this information from each of the neighbors can be of interest since we are ultimately interested in computing only $F(u_i) \cap R_t(u_i)$. If the local neighborhood has the same topology at each processor, the information to be transmitted by a processor to each of its neighbors can easily be evaluated, using a syndrome decoding function.

As an example, consider the hexagonal grid interconnection with a local neighborhood as shown in Fig. 1. Since the topology of the local neighborhood is the same, and each processor executes a copy of the same algorithm, let the processor under consideration be simply labeled u with the t neighbors of the processor labeled from 1 to t. It is clear that the central processor must route the test outcome corresponding to its neighbor 1 to neighbor 4, the test outcome corresponding to its neighbor 2 to neighbor 5, and so on, as illustrated in Fig. 1. In other words, the syndrome routing function f is given by $f(k) = (k + 2)\ mod\ 6 + 1$. Clearly, a syndrome routing function will be one-to-one and onto on the set of neighbors and will be dependent on the relative ordering of the adjacent processors and the chosen local neighborhood.



$f(1) = 4$, $f(2) = 5$, $f(3) = 6$,
$f(4) = 1$, $f(5) = 2$, $f(6) = 3$.

Fig. 1. A local neighborhood of order 6 in a hexagonal grid and a syndrome routing function.

In order to implement the algorithm in a distributed manner, we assume the existence of two t-bit registers A and B at each processor. $A[k]$ contains the comparison outcome available at that processor corresponding to the kth neighbor. In other words, $A[k] = 1$ if, and only if, k belongs to $N_1(u)$ at processor u. The register B will be used to receive information from the neighbors. Let $B[k]$ correspond to the comparison information sent by the kth neighbor. Now, observe

Consider the following syndrome: $a_{n1} = 1$, $a_{12} = 1$, $a_{23} = 1$, $a_{34} = 1$, $a_{45} = 1$ and all other outcomes are 0. The fault sets $F_1 = \{u_1, u_3, u_4\}$ and $F_2 = \{u_1, u_2, u_4\}$, two permissible fault sets under the given fault constraint, are allowed faults for this syndrome.

Case 2.2. $p = 2$, $q = 4$.

Since n is even, let F be a fault set containing alternate processors in S. Then the fault sets F and F^c are allowable fault sets for the syndrome in which all outcomes are 1.

Case 2.3. $p = 2$, $q \geq 5$.

We note that if there are at most 2 faulty processors in any consecutive 5 processors then for any processor u, $L_2(u) \cup \{u\}$ which consists of 5 processors contains at most 2 faulty processors. Thus, given a permissible syndrome, the local diagnosis algorithm developed in the previous section can be used to identify all processors correctly. Since the constraint $p = 2$ and $q = 5$ permits all fault sets which are valid when $p = 2$ and $q \geq 5$, these values for p and q admit the maximum number of fault sets which can be uniquely diagnosed. □

If a fault constraint permits a fault set F and its complement F^c to be permissible fault sets, then given a valid syndrome, the faulty processors may not be correctly identified; the fault sets F and F^c generate a common syndrome. We note that if initially one processor v is correctly determined to be fault-free or less than half the total number of processors in the system are faulty, then for any subset F of U, at most one of the subsets F and F^c can be an allowable fault set for a given syndrome.

Theorem 3. Let S be a ring of n processors in which one of the following conditions is satisfied:

(1) some processor is known to be fault-free and $n \geq 5$
(2) less than half the processors in the system are faulty and $n \geq 7$
(3) n is odd.

Then the values for p and q, which permit the maximum number of fault sets which can be uniquely diagnosed in S under the local constraint of at most p faulty processors in any q consecutive processors, are $p = 2$ and $q = 4$ respectively.

Proof. It can be verified as in the proof of Theorem 2 that the case $p \geq 3$ and the case $p = 2$ and $q = 3$ may result in the syndromes which cannot be uniquely diagnosed.

Assume $p = 2$ and $q = 4$. We first show that if one fault-free processor v is known to be fault-free, all other processors can be identified correctly. We assume that the diagnosis procedure initiated at v proceeds clockwise. If a processor is fault-free then the adjacent processor can be correctly identified. If two consecutive processors are identified as faulty then the next processor can be correctly identified as fault-free. Thus, only the situation shown in Fig. 2 could pose a problem.

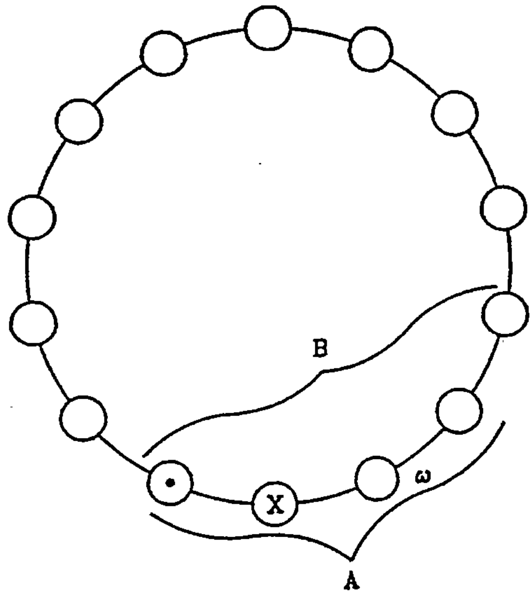


Fig. 2. Illustration for Local Diagnosis Algorithm in a ring of processors.

Since there are at most 2 faulty processors in any 4 consecutive processors, there are at most 2 faulty processors in A. Hence, B contains at most 2 faulty processors. The processor w has at most 2 faulty processors in its local neighborhood $L_2(w) \cup \{w\}$. Thus, again the syndrome decoding algorithm developed in the previous section can be carried out with respect to processor w to determine its status. Thus, if one processor is given to be fault-free or can be identified correctly to be fault-free, then all other processors can be identified correctly.

We now show how one processor can be identified correctly if either (2) or (3) is true. We note that if a valid syndrome contains the sequence of consecutive outcomes 00, 011 or 110 then the processors adjacent to the 0-links are fault-free; otherwise there is a sequence of 4 consecutive processors of which at least three are faulty.

We now claim that for a valid syndrome, one of the following sequences of outcomes 00, 011 or 110 occurs. Assume the contrary. Then the following syndromes are the only syndromes which do not contain any of the sequences 00, 011 or 110; the syndrome s_1 in which all outcomes have value 1 and the syndrome s_2 in which 0 and 1 outcomes alternate.

Case 1. Less than half the processors are faulty and $n \geq 7$.

In this case S contains two consecutive fault-free processors. Hence, there exists at least one 0-link and the syndrome s_1 cannot occur. Since at most 2 of any 4 consecutive processors can be faulty, the syndrome s_2 corresponds to fault sets in which two faulty processors are followed by two fault-free processors and vice versa. But this contradicts the assumption that the number of faulty processors is less than the number of fault-free processors.

that $A[k]$ XOR $B[k] = 1$ if, and only if, the corresponding neighbor of u's kth neighbor belongs to the intersection of $N_0(N_1(u)) \cup N_1(N_0(u))$ and $R_t(u)$. Thus the computation of $|F(u) \cap R_t(u)|$ can be carried out very efficiently through simple logical operations at each processor. Clearly, this syndrome decoding can be done in constant parallel time.

As stated in the introduction the development of the local diagnosis criterion has been motivated primarily from the viewpoint of its application to regular interconnection structures such as rectangular hexagonal and octagonal grids with wraparound in two dimensional structures, binary n-cube connected cycles, and hypercube connections. In these architectures each processing element is connected to the same fixed number of other processing elements with perfect symmetry with respect to interconnections. Thus, it is easy to observe that each of these architectures permits the construction of local neighborhoods of order t around each processor, for t is equal to the number of neighbors of any processor.

We wish to note that for regular interconnected structures, the local diagnosis criterion developed in this paper permits the correct diagnosis of fault patterns which cannot be handled by the classical t-diagnosis. However, the question still remains as to whether or not a local neighborhood of order t can be constructed around every unit in an arbitrary t-diagnosable system so that the local diagnosis approach can be used. Unfortunately, this is not so. But it has been shown in [12] that every system S for which a local neighborhood of order t can be defined around each processor is t-diagnosable under the comparison model.

4. Diagnosis of a Ring of Processors under Local Fault Constraints

In this section, we analyze the implication of imposing local fault constraints on a ring of processors. Specifically, we determine if, given a syndrome, we can uniquely determine the set of faulty processors as long as at most p out of any q consecutive processors are faulty.

Theorem 2. Let S be a ring of n processors where n is even. Given that at most p processors are faulty out of any q consecutive processors, the values for p and q which admit the maximum number of fault sets that can be uniquely diagnosed are $p = 2$ and $q = 5$.

Proof. Let $\{u_1, u_2, \ldots, u_n)$ be the ring processors.

Case 1. $3 \leq p \leq n$.

We show that in this case, the set of permissible fault sets cannot be uniquely diagnosed. Consider the following syndrome: $a_{n1} = 1$, $a_{12} = 1$, $a_{23} = 1$, $a_{34} = 1$ and all other outcomes have value 0. Both $F_1 = \{u_1, u_2, u_3\}$ and $F_2 = \{u_1, u_3\}$ are allowable fault sets for this syndrome.

Case 2. $p = 2$, $q \geq 3$.

Case 2.1. $p = 2$, $q = 3$.

Case 2. n is odd.

In this case, the syndrome s_2 cannot be present. Since at most 2 out of any 4 consecutive processors can be faulty, the syndrome s_1 corresponds to fault sets in which faulty and fault-free processors alternate; this is not possible since n is odd and S is a ring of processors.

This shows that one fault-free processor can be determined if either (2) or (3) is true.

□

The diagnosis algorithms outlined in the course of the proofs of Theorem 2 and Theorem 3 can be executed in a sequential manner on a host processor. As regards distributed implementation, consider first the case when the ring has an even number of processors. In this case, each processor executes the syndrome decoding algorithm in parallel and determines its status (faulty or fault-free) in constant time. In the case when the ring has an odd number of processors, diagnosis involves two phases. In the first phase, each processor checks to see if it is adjacent to a 0-link which lies on a path defined by consecutive outcomes 00, 011 or 110. If so, then the processor declares itself fault-free. This phase can also be executed in parallel in constant time. In the second phase, a fault-free processor initiates the diagnosis algorithm (as described in the proof of Theorem 2). In this phase, a processor may have to execute the syndrome decoding algorithm to determine its status. If more than one fault-free processor are identified in the first phase, then all these processors can initiate phase 2 simultaneously. In the worst case, phase 2 may take $O(n)$ time.

It can be easily seen that the message complexity of the above distributed diagnosis algorithms for a ring is $O(n)$. We wish to note that these distributed algorithms can be made to run on both bidirectional and unidirectional rings.

5. Summary and Conclusions

In this paper we have studied fault diagnosis of a ring of processors. We have used the comparison based model of Chwa and Hakimi [4]. Based on Theorem 1, we have shown that the processors in a ring can be uniquely diagnosed as faulty or fault-free if, and only if, the fault pattern satisfies certain conditions. More specifically, a ring of n processors can be uniquely diagnosed if, and only if, there are at most two faults in any consecutive five processors where n is even, or there are at most two faults among any consecutive four processors if n is odd. Proofs of these results are constructive and lead to diagnosis algorithms which permit distributed implementation. It has been shown in [11] that the maximum value of t for which most regular interconnected systems are t-in-L_1 diagnosable is given by $t = \frac{\Delta}{2} + 1$, where Δ is the node degree. It follows from the proofs of Theorems 2 and 3 that the ring for which $\Delta = 2$ is not 2-in-L_1 diagnosable. Thus, our characterization of syndromes which permit unique diagnosis of a ring of processors and the resulting distributed diagnosis algorithms advance the theory of local diagnosis of multiprocessor systems presented in [11].

References

[1] F. P. Preparata, G. Metze and R. T. Chien, On the connection assignment problem of diagnosable systems, *IEEE Trans. Electr. Comput.* EC-16 (1967) 848–854.

[2] S. L. Hakimi and A. Amin, Characterization of the connection assignment of diagnosable systems, *IEEE Trans. Comput.* C-23 (1974) 86–88.

[3] S. Karunanidhi and A. D. Friedman, Analysis of digital systems using a new measure of system diagnosis, *IEEE Trans. Comput.* C-28 (1979) 121–133.

[4] K. Y. Chwa and S. L. Hakimi, Schemes for fault-tolerant computing: A comparison of modularly redundant and t-diagnosable systems, *Inform. Contr.* 49 (1976) 585–593.

[5] A. T. Dahbura and G. M. Masson, An $O(n^{2.5})$ fault identification algorithm for diagnosable systems, *IEEE Trans. Comput.* C-33 (1984) 486–492.

[6] G. Sullivan, An $O(t^3 + |E|)$ fault identification algorithm for diagnosable systems, *IEEE Trans. Comput.* C-37 (1988) 388–397.

[7] D. Fussell and S. Rangarajan, Probabilistic diagnosis of multiprocessor systems with arbitrary connectivity, in *19th Int. Symp. on Fault-Tolerant Comp.* (IEEE Comp. Soc. Press, 1989) 560–565.

[8] S. Rangarajan and D. Fussell, Probabilistic diagnosis tailored to system topology, in *21st Int. Symp. on Fault-Tolerant Comp.* (IEEE Comp. Soc. Press, 1991) 230–237.

[9] P. Berman and A. Pelc, Distributed probabilistic fault diagnosis for multiprocessor systems, in *20th Int. Symp. on Fault-Tolerant Comp.* (IEEE Comp. Soc. Press, 1991) 340–346.

[10] A. K. Somani and V. K. Agarwal, Distributed syndrome decoding for regular interconnected structures, in *19th Int. Symp. on Fault-Tolerant Comp.* (IEEE Comp. Soc. Press, 1989).

[11] A. Das, K. Thulasiraman, K. B. Lakshmanan and V. K. Agarwal, Multiprocessor fault diagnosis under local constraints. Accepted for publication in *IEEE Trans. Comput.*

[12] A. Das, Diagnosability and diagnosis of sparsely interconnected multiprocessor systems, Ph.D. Dissertation, Concordia University, Montreal, 1989.