An Efficient Placement and Routing Technique for Fault-tolerant Distributed Embedded Computing

Roozbeh Jafari, Foad Dabiri, Majid Sarrafzadeh
Computer Science Department
University of California
Los Angeles, CA 90095
{rjafari,dabiri,majid}@cs.ucla.edu

Abstract

This paper presents an efficient technique for placement and routing of sensors/actuators and processing units in a grid network. The driver application that we present is a medical jacket which requires an extremely high level of robustness and fault tolerance. The power consumption of such jacket is another key technological constraint. Our proposed interconnection network is a mesh of wires. A jacket made of fabric and wires would be susceptible to accidental damage via tears. By modeling the tears, we evaluate the probability of having failures on every segment of wires in our mesh interconnection network. Then we study two problems of placement and routing in the sensor networks such that the fault tolerance is maximized while the power consumption is minimized. We develop efficient integer linear programming (ILP) formulations to address these problems and perform both placement and routing simultaneously. This ensures that the solution is a lower bound for both problems. We evaluate the effectiveness of our proposed techniques on a variety of benchmarks.

1 Introduction

The past few years have seen exciting advances in the development of pervasive computing technologies. Computation, storage, and communication are now more or less woven into the fabrics of our society with much of the progress being due to the relentless march of Silicon-based electronics technology as predicted by Moore's Law. The emerging field of flexible electronics, where electronic components such as transistors and wires are built on a thin flexible material, offers a similar opportunity to weave computation, storage, and communication into the fabric of the very clothing that we wear, thereby creating an intelligent fabric (also called electronic textiles or e-textiles) [4]. Electronic components built of flexible materials have characteristics that are very different from that of silicon and PCB-based electronics. Fault tolerance become more eminent in this architecture due to criticality of the applications running on this architecture. Further, the operating scenarios of these systems involve environmental dynamics, physical coupling, resource constraints, infrastructure support, and robustness requirements that are distinct from those faced by traditional systems. This unique combination requires one to go beyond thinking of these systems as traditional electronic systems in a different form factor. Instead, a re-thinking and complete overhaul of the system architecture and the design methodology for all layers of these systems is required.

1.1 Driver Application

We use pervasive patient monitoring and sensor-driven personalized trans-dermal drug delivery as our driver application [3]. One possibility of leveraging electronic textile technology in the context of such an application is to create a flexible garment (i.e., vest) that the patient can wear, which has sensing, computation, communication, and actuation elements embedded in it. We are developing such a prototype vest called medical jacket. Ideally, such a personalized drug delivery vest should have sensors on both the interior (to measure physiological readings) and on the exterior (to measure environmental readings such as the presence of toxins in the surroundings), and a software-controlled, electrically-actuated trans-dermal drug delivery system. It should allow low-latency, fine-grained adaptation of the drug dosage based on continual physiological measurements in the case of patients, and based on both environmental and physiological measurements in the case of people operating in hazardous environments. More generally, our application driver is representative of biomedical applications where infor-

Proceedings of the 11th IEEE International Conference on Embedded and Real-Time Computing Systems and Applications (RTCSA'05)
1533-2306/05 $20.00 © 2005 IEEE

mation technology is integrated into fabrics and textiles.

2 Related Work

Placement and routing problems on a grid have been studied extensively in the field of VLSI CAD. There has been considerable research effort to solve these problems. Like most other VLSI layout problems, it is believed that placement and routing cannot be solved optimally in a reasonable amount time [8] [7]. Therefore, heuristic algorithms are used to obtain near-optimal solutions.

The main difference between our approach and other existing techniques is that we perform both placement and routing simultaneously. The solution that we find is also a lower bound for both problems. Our technique easily accommodates our requirements in terms of the instance size and the number of sensors, actuators and processing units to be placed and routed.

Message routing in large interconnection networks has received a great deal of attention in recent years in the field of parallel computing. To decrease the amount of time in transmitting data, second generation multicomputers adopt wormhole routing mechanism, e.g., Ametek 2010, nCUBE-2, Intel Paragon, J-machine, and iWARP [5]. Adaptive wormhole routing algorithms have been proposed for mesh, torus, and hypercube topologies. With wormhole routing, a message is divided into a sequence of fixed-size units of data, called 'flits'. The header it of a message contains all the information needed to decide the selection of next channel on the route. As the header flit advances, the remaining flits follow it in a pipeline fashion. When the header flit reaches a node that has no suitable output channel available, all of the flits in the message are blocked until the channel is freed. This form of routing technique makes the message transmission time almost independent of the distance between two nodes if the network is contention-free. A good survey paper on wormhole routing techniques can be found in [5]. The main difference between our approach and wormhole routing is that we employ the concept of circuit switching for communication while wormhole routing utilized packet switching technique. In our model, since we have continuous data collection from sensors, we employ the concept of circuit switching which provides dedicated communication line between nodes. Barrenechea et al. studied optimal routing algorithms for regular sensor networks, namely, square and torus grid sensor networks, in both, the static case (no node failures) and the dynamic case (node failures) [1]. Their model, however, utilizes packet switching approach.

Multihop routing in wireless networks with respect to power optimization and fault-tolerance have been studied extensively. Generally speaking, a node in a wireless network independently explores its surrounding region and establishes connections with other neighboring nodes that are within its transmission and reception range. Our model significantly differs since the interconnect network is not via wireless media. The wired interconnect networks reduces the environmental effects on the reliability of the network and greatly reduces the communication power consumption.

3 Preliminaries

3.1 Interconnection Topology

The interconnection medium for our proposed system is a mesh of wire segments. The mesh interconnection topology is a wire-frame that has a regular structure, each vertex being connected to exactly four other vertices. Mesh networks have several significant advantages. Each node has a dedicated communication link with every other node on the network and also has access to the full bandwidth available for that link. Nodes on buses must share the bandwidth available on the bus medium. Besides, in a mesh, multiple paths exist between devices. This brings a great robustness against faults. If a direct path between two nodes goes down, messages can be rerouted through other paths. Moreover, it has considerable scalability and can be easily manufactured. The manufacturing issues become more significant because in our system wires are integrated into the fabric and this can be easily achieved with the current fabric manufacturing technology. Furthermore, the mesh interconnection is highly regular which assists us in routing and placement of sensors and processing units.

In the remainder of the paper, we refer to mesh interconnection topology by terminology "grid". The interconnection topology is modeled as a grid graph illustrated in Figure 1. Each edge in the grid represent a wire or a communication link and each node represents a processing unit and a switch. Most nodes are connected to four edges. Switches are responsible for either connecting the processing units to any of the edges or establish the connections between edges. In this paper we focus on routing and placement of sensors and processing units. We emphasis on finding disjoint paths between every pair of communicating nodes because our communication medium is a mesh of wires and each wire cannot handle more than one communication at a time. In addition, we chose the wired interconnection medium due to high available bandwidth of wires and the requirements of our application (Electrocardiogram signals transmit at 115.2Kbps in our application).

Proceedings of the 11th IEEE International Conference on Embedded and Real-Time Computing Systems and Applications (RTCSA'05)
1533-2306/05 $20.00 © 2005 IEEE

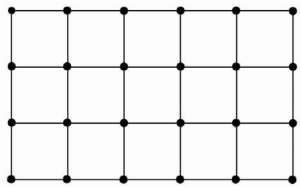

Figure 1. Interconnection Topology

3.2 Switch Model

Our proposed switch model is shown in Figure 2. Our switches are capable of connecting any terminals to any other terminals.

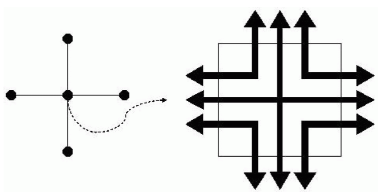

Figure 2. Switch Model

3.3 Power Lifetime

The mobility of our system requires the use of constrained power sources (i.e. batteries). Therefore, in the design of such system, power optimization is one of the major objectives. The amount of power resources available to the system determines the life time of the system which will be referred to as power lifetime (l_p) in this paper. As shown in the following equations, E indicates the remaining energy of the system power source while P_w represents the power consumption rate of each node in the grid which includes the power consumption of both the processing unit and the switch.

$$\begin{aligned} l_p &= \text{System power lifetime} \\ E &= \text{Energy remaining in the system power source} \\ P_w &= \text{power consumption of each active node} \end{aligned}$$

V is a set of all the nodes of the grid while R_v consists of the nodes chosen and utilized for routing. Moreover, the size of set R_v is denoted by n.

$$V = \{v_i | v_i \text{ is a node in the grid}\}$$

$$R_v = \{v_i | v_i \in V \wedge v_i \text{ used in routing}\}$$

$$n = |R_v|$$

Finally, we define the power lifetime of the system as follows:

$$l_p = \frac{E}{P_w n} = \frac{1}{\lambda_p n} \text{ where } \lambda_p = \frac{P_w}{E} \tag{1}$$

We also define λ_p which will be used in the ILP problem formulation of routing and placement.

3.4 Power Consumption Model

In our proposed architecture, it is crucial to minimize the power consumption since the system is battery-powered. The power consumption due to processing is rather constant as long as the processing units remain operational. We, however, can reduce the power consumption of inter-node communications. When a communication between a pair of nodes is established, a number of switches is utilized. The power consumption of the communication is proportional to the number of switches along the path considering that they must remain operational throughout the communication. The other important factor which is generally considered is the delay. However, in our system, the communications are established using circuit switching scheme. Therefore, the wire length of each path is rather small and delay concerns become a non-issue.

3.5 Fault Lifetime

As discussed in Section 3.1, our system is made of fabric which is susceptible to accidental damage via tears and punctures. Such faults may yield loss of a connection and therefore, may result in total system failure. Firefighters, policemen, soldiers, astronauts, athletes working in hazardous environments may experience more faults than others. We define λ_f as the rate of fault occurrence per unit time. We define E and R_e as set of edges in the grid and set of edges used in routing, respectively:

$$\begin{aligned} E &= \{e_i | e_i \text{ is an edge in the grid}\} \\ R_e &= \{e_i | e_i \in E \wedge e_i \text{ used in routing}\} \end{aligned}$$

The set of edges not used in routing is defined by R'_e:

$$\begin{aligned} R'_e &= E - R_e \\ &= \{e_i | e_i \in E \wedge e_i \text{ NOT used in routing}\} \end{aligned}$$

When a fault occurs, probability of loosing a particular edge e_i is represented by $P(e_i)$.

$$P(e_i) = \Pr\{e_i \text{ fails} | \text{a fault has occurred}\} \tag{2}$$

Assuming that the routing is performed (i.e. the edges for inter-communication between nodes are selected), when a fault happens, the probability of having system failure due to link failures is depicted by $P(e_R)$.

$$\begin{aligned} P(e_R) &= \Pr\{\text{an edge in } R_e \text{ fails} | \text{a fault has occurred}\} \\ &= \textstyle\sum_{e_i \in R_e} P(e_i) = \alpha \end{aligned}$$

Obviously, failures in $e_i \in R'_e$ does not affect the system functionality since $e_i \in R'_e$ is not utilized for

Proceedings of the 11th IEEE International Conference on Embedded and Real-Time Computing Systems and Applications (RTCSA'05)

routing. The probability of not having system failure when a fault occurs is shown with $P(e_{R'})$.

$$P(e_{R'}) = 1 - P(e_R) = 1 - \alpha$$

The expected fault lifetime of the system now can be defined in the following equation. The first term in the series indicates the probability of having system failure after the occurrence of the first fault. The second term denotes the probability of system failure with two faults while the first fault refrains the system intact. The same arguments hold for all the remaining terms.

$$\begin{aligned} \overline{l_f} &= \text{Expected lifetime of the system} \\ &= \tfrac{\alpha}{\lambda_f} + \tfrac{2(1-\alpha)\alpha}{\lambda_f} + \tfrac{3(1-\alpha)^2\alpha}{\lambda_f} + \cdots \\ &= \tfrac{1}{\alpha\lambda_f}\sum_{i=1}^{\infty} i(1-\alpha)^{i-1} \end{aligned}$$

The expected fault lifetime is shown by Equation 3 where λ_f and α are the fault occurrence rate and probability of having system failure respectively.

$$\overline{l_f} = \tfrac{\alpha}{\lambda_f(1-(1-\alpha))^2} = \tfrac{1}{\lambda_f \alpha} \tag{3}$$

3.6 Fault Model

In this section we model 'tears' or 'cuts' that may occur to the fabric and yield loss of a connection in the network. For simplicity, we consider only monotone tears. In practice, this assumption is not far from reality because the results from tear and puncture analysis experiments on textile confirm that the nature of tears is usually monotone (WARP and WEFT tears) [9]. Various mechanical instruments are being used for these experiments such as Elmendorf (falling pendulum) [2] and COMPUTE (COMbined PUncture and TEar) [6] on cotton fabric. In addition, several standards have been implemented for tear analysis on textile such as PE-P-04640:1976 and PE-EN ISO 13937-3:2002:1-4.

Unfortunately most reports published in this area only describe statistics about the strength/length of tears and not about their physical distribution throughout the fabric. Therefore, the following theoretical tear analysis was carried out with the assumptions based on the outcome of practical tear analysis experiments.

Each tear can be modeled as a path between two nodes in the dual graph of the grid as shown in Figure 3. The dual graph refers to the graph in which each region of the grid is presented with a node. Whenever two regions in the grid have an edge in common, there exists an edge between corresponding nodes in the dual graph. According to this model, the total

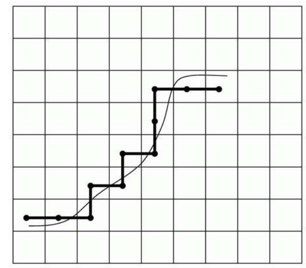

Figure 3. Representation of Tears in Dual Graph

number of different tears that may happen in the fabric is the total number of distinct paths in the dual graph. Therefore, the number of distinct tear occurrence on a wire segment in the grid is the same as the number of paths going through the corresponding edge in the dual graph. Consequently, we must count the number of paths which passes through each edge in the dual graph. To show how this is performed, initially, we present the following lemmas.

Lemma 3.1 *The number of distinct paths from node v_{11} to v_{nm} on a grid is:*

$$P(v_{11}, v_{nm}) = \binom{n+m-2}{m-1} \tag{4}$$

Proof: Length of each path from v_{11} to v_{nm} is $m+n-1$ and each path can be represented with a string of length $m+n-2$ containing of right (R) and up (U) moves. Each string has exactly $n-1$ (R)'s. The total number of distinct strings is shown in (4) which is equivalent to the number of paths.

Now, consider an edge w which connects nodes

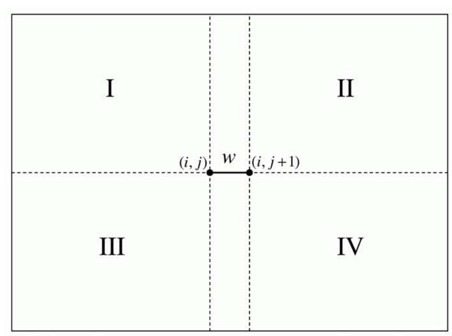

Figure 4. Region Subdivisions for every Edge in Dual Graph

v_{ij} to v_{ij+1} as shown in Figure 4. Since we consider only monotone tears, the total number of paths which passes through w, $T(w)$, can be evaluated by (5):

$$\begin{aligned} T(w) &= R(I, v_{ij}) \times R(IV, v_{ij+1}) \\ &+ R(III, v_{ij}) \times R(II, v_{ij+1}) \end{aligned} \tag{5}$$

$R(X, v_{ij})$ is the total number of paths connecting every node in rectangle X to node v_{ij}. Possible rectangles for X are shown in Figure 4. We have:

$$R(I, v_{i,j}) = \sum_{s=1}^{i} \sum_{t=1}^{j} P(v_{st}, v_{ij})$$

To find a closed form for $R(I, v_{i,j})$ we observe that

$$\begin{aligned} \textstyle\sum_{s=1}^{j} P(v_{s1}, v_{ij}) &= \textstyle\sum_{s=1}^{j} \begin{pmatrix} j-1+i-s \\ j-1 \end{pmatrix} \\ &= \begin{pmatrix} j+i-1 \\ j \end{pmatrix} \end{aligned}$$

This implies that the total number of paths originating from every node on the left border of rectangle I to v_{ij} is equal to $P(v_{11}, v_{ij+1})$.
Finally we conclude that:

$$R(I, v_{ij}) = \sum_{t=1}^{j} \begin{pmatrix} j+t-1 \\ t \end{pmatrix} = \begin{pmatrix} i+j \\ j \end{pmatrix} \quad (6)$$

Similar equations hold for $R(IV, v_{ij+1})$, $R(III, v_{ij})$, $R(II, v_{ij+1})$. Note that we only considered horizontal edges. The same argument is valid for vertical edges. We previously showed that each tear in the grid is represented by a path in the corresponding dual graph. Therefore, assuming that a tear occurs, the probability of loosing an edge in the grid is proportional to the number of distinct paths in the dual graph crossing that edge. (i.e. $T(w)$). The probability of having failure in edge e_k is calculated as follows:

$$Pr(e_k) = \frac{T(w)}{\sum T(w_i)\ (\text{for} \forall e_i \in E)} \quad (7)$$

In Equation 7, each $w_i \in$ dual graph corresponds to $e_i \in$ in the grid. Figure 5 illustrates the probability of failure of each edge in the grid when a tear occurs.

In the model we illustrated, our analysis show that some wires are more susceptible to failures. If the wires already utilized for communication fail, the system will fail. Furthermore, since the lifetime of the system is determined by the shortest lifetime of all wire segments selected for routing, we prefer to employ wires which are less susceptible to faults.

4 Problem Definition

We introduced the terminologies power lifetime and fault lifetime as well as the tear analysis technique. The lifetime of our system is highly dependent on the remaining energy in the power source and the frequency of fault occurrence. Sometimes the system may fail due to lack of energy and in some cases, faults may parallelize the system. To increase the power lifetime, once we perform placement and routing, we seek to minimize the total length of routing. To improve the fault lifetime, we attempt to use the edges which are less susceptible to faults. The overall lifetime of the system is dependent both on power and fault lifetime. Therefore, to maximize the overall lifetime of the system, we shall maximize the minimum of power and fault lifetimes.

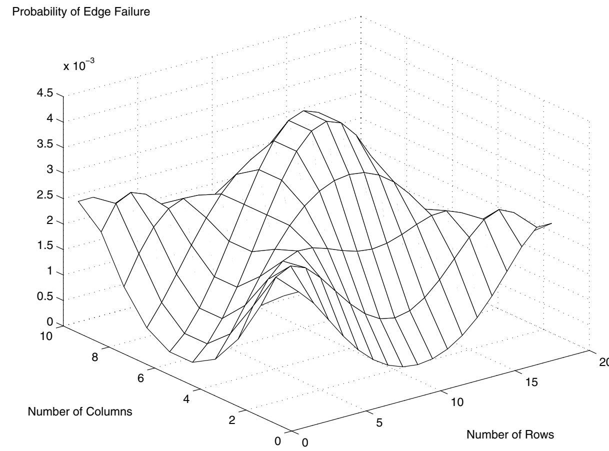

Figure 5. Probability of Edge Failure in the Grid for all Possible Distinct Tears

$$MAX(MIN(l_p, \overline{l_f})) \quad (8)$$

However, the objective function in an ILP formulation may not contain both MAX and MIN as shown in (8). Therefore, we redefine the objective functions as illustrated in (9) and (10) and have the solver optimize both objectives individually in two different runs.

$$\begin{array}{c} MAX(l_p) \\ \text{with constraint } l_p > \overline{l_f} \end{array} \quad (9)$$

$$\begin{array}{c} MAX(\overline{l_f}) \\ \text{with constraint } \overline{l_f} > l_p \end{array} \quad (10)$$

The maximum value of (9) and (10) is the value of objective function (8). In Section 5 we specify in details how the objective functions are constructed.

5 Problem Formulation

5.1 Routing

In both placement and routing problems, we assume that the interconnection network is a grid of size $M \times N$. The vertices correspond to processing units along with switches while the edges corresponds to the links between them. Each vertex and edge in the grid is assigned a variable. These variables specify if an edge and/or vertex is selected for routing and/or placement. Further details on how these variables are

Proceedings of the 11th IEEE International Conference on Embedded and Real-Time Computing Systems and Applications (RTCSA'05)
1533-2306/05 $20.00 © 2005 IEEE

used will be explained later. However, the accordance between the indices of variables and the position of vertices and/or edges are as follows:
Variable x_{pij} represent if node with coordinations i and j in the grid is chosen for the pth communication pair. As shown in Figure 6, variables y_{hij} and y_{vij} represents horizontal and vertical edges corresponding to node ij in the grid, respectively.

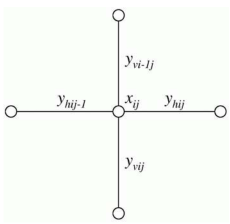

Figure 6. ILP Variables

Problem: *Routing to Minimize the Power Consumption*
Instance: *Graph $G(V,E)$ s.t. G is a grid, collection of disjoint vertex pairs $(s_1,t_1),(s_2,t_2),...,(s_l,t_k)$*
Question: *Find k mutually vertex-disjoint paths in G, connecting s_i to t_i for each i, $1 \leq i \leq k$ s.t. the total length of paths is minimum.*

In the ILP formulation we define variables x_{pij} where $p = 1,...,k$. x_pij represents whether vertex v_{ij} is selected as a part of the path between (s_p,t_p). We also define q_{pij} as follows:

$$x_{pij} = \begin{cases} 1, & \text{if vetex } v_{ij} \text{ is selected for} \\ & \text{communicating pair } (s_p,t_p) \\ 0, & \text{otherwise.} \end{cases} \quad (11)$$

$$y_{(h/v)pij} = \begin{cases} 1, & \text{if edge } y_{(h/v)pij} \text{ is selected for} \\ & \text{communicating pair } (s_p,t_p) \\ 0, & \text{otherwise.} \end{cases} \quad (12)$$

$$q_{pij} = \begin{cases} 1, & \text{if vertex } v_{ij} = s_p \vee v_{ij} = t_p \\ 0, & \text{otherwise.} \end{cases} \quad (13)$$

Using the above variables, we define the constraints (14), (15) and (16) for the ILP formulation. (14) ensures that a feasible path is found between s_p and t_p while (15) denotes that an each edge in the grid may not be used for more than one path. (16) enforces that the paths are vertex disjoint. To ensure that a path is feasible, we enforce the condition that exactly two neighboring edges of a vertex must be selected if the vertex is chosen for routing. In the case where the vertex is not chosen, none of the neighboring edges may be selected. This condition is slightly different for the source and sink vertices of the path since exactly one neighboring edge must be selected for routing. This condition is expressed in (14).
Finally the objective function, represented in (17), attempts to minimize the power consumption or the number of nodes employed for routing. Subsequently, this would maximize the power lifetime.

$$\begin{array}{c} 2x_{pij} - y_{phij-1} - y_{pvi-1j} - y_{phij} - y_{pvij} - q_{pi} = 0 \\ 1 \leq p \leq k \\ 1 \leq i \leq M \\ 1 \leq j \leq N \end{array} \quad (14)$$

$$\sum_{p=1}^{k} y_{p(h/v)ij)} \leq 1 \quad (15)$$

$$\sum_{p=1}^{k} x_{pij)} \leq 1 \quad (16)$$

$$MIN(\sum_{p=1}^{k}\sum_{i=1}^{m}\sum_{j=1}^{n} x_{pij}) \quad (17)$$

Problem: *Routing to Maximize the Fault Tolerance*
Instance: *Graph $G(V,E)$ s.t. G is a grid, fault occurrence weights on edges of G, collection of disjoint vertex pairs $(s_1,t_1),(s_2,t_2),...,(s_l,t_k)$*
Question: *Find k mutually vertex-disjoint paths in G, connecting s_i to t_i for each i, $1 \leq i \leq k$ s.t. (18) is minimum.*

Tear analysis described in Section 3.6 generates probability of having failures in every edge of the grid once a tear occurs. This information helps us to improve the fault tolerance of the system. As opposed to the previous problem, the objective function is to pick edges for routing such that the probability of fault occurrence in the selected edges is minimized. Therefore, the constraints (14), (15) and (16) still hold while the objective function (17) changes according to the new requirements.

$$MIN(\sum_{p=1}^{k}\sum_{i=1}^{m}\sum_{j=1}^{n} P(e_{hij})y_{hpij} + P(e_{vij})y_{vpij}) \quad (18)$$

Problem: *Routing to Maximize the Fault Tolerance and Minimize the Power Consumption*
Instance: *Graph $G(V,E)$ s.t. G is a grid, fault occurrence weights on edges of G, collection of disjoint vertex pairs $(s_1,t_1),(s_2,t_2),...,(s_l,t_k)$*
Question: *Find k mutually vertex-disjoint paths in G, connecting s_i to t_i for each i, $1 \leq i \leq k$ s.t. (19) and/or (20) is minimum.*

The first problem that we discussed earlier in Section

Proceedings of the 11th IEEE International Conference on Embedded and Real-Time Computing Systems and Applications (RTCSA'05)
1533-2306/05 $20.00 © 2005 **IEEE**

5.1 addresses the issue of power optimization. It attempts to minimize the total number of nodes selected for routing. Since selecting each node in the grid requires activation of a switch and a processing unit, therefore the objective function minimizes the power consumption of the network. The previous problem, however, denotes how we can improve the fault tolerance of our interconnection medium through using edges which are less exposed to faults. Now we present a new problem formulation which addresses both power optimization and fault tolerance issues simultaneously. The constraints (14), (15) and (16) are still valid, however, the objective functions (19) and (20) determine the lower bound for the lifetime of our system as described in Section 4.

$$\begin{gathered} MIN(\sum_{p=1}^{k}\sum_{i=1}^{m}\sum_{j=1}^{n} x_{pij}) \\ \text{with respect to} \\ \sum_{p=1}^{k}\sum_{i=1}^{m}\sum_{j=1}^{n} x_{pij} > \\ \sum_{p=1}^{k}\sum_{i=1}^{m}\sum_{j=1}^{n} P(e_{hij})y_{hpij} + P(e_{vij})y_{vpij} \end{gathered} \tag{19}$$

$$\begin{gathered} MIN(\sum_{p=1}^{k}\sum_{i=1}^{m}\sum_{j=1}^{n} P(e_{hij})y_{hpij} + P(e_{vij})y_{vpij}) \\ \text{with respect to} \\ \sum_{p=1}^{k}\sum_{i=1}^{m}\sum_{j=1}^{n} P(e_{hij})y_{hpij} + P(e_{vij})y_{vpij} > \\ \sum_{p=1}^{k}\sum_{i=1}^{m}\sum_{j=1}^{n} x_{pij} \end{gathered} \tag{20}$$

The next problem that we present is an extension to the previous problem where each edge in the grid carries more than one wire. We define the channel capacity for each edge in the grid and assume that each edge contains c wires. Thus, at most c disjoint paths may use one edge simultaneously.

Problem: *Routing to Maximize the Fault Tolerance and Minimize the Power Consumption*
Instance: *Graph $G(V, E)$ s.t. G is a grid, fault occurrence weights on edges of G, collection of disjoint vertex pairs $(s_1, t_1), (s_2, t_2), ..., (s_l, t_k)$*
Question: *Find k mutually vertex-disjoint paths in G, connecting s_i to t_i for each i, $1 \le i \le k$ s.t. each edge in E is not associated with more than c paths and (19) and/or (20) is minimum.*

With our new assumption, constraint (14) is still effective while (21) and (22) are enforced as well. The objective function remains the same as (19) and (20).

$$\sum_{p=1}^{k} y_{p(h/v)ij)} \le c \tag{21}$$

$$\sum_{p=1}^{k} x_{pij)} \le c \tag{22}$$

The last ILP formulation for routing is for the case where a processing unit is connected to multiple sensors. This case happens where multiple sensors communicate with a single processing unit simultaneously and efficiently.

Problem: *Routing to Maximize the Fault Tolerance and Minimize the Power Consumption*
Instance: *Graph $G(V, E)$ s.t. G is a grid, fault occurrence weights on edges of G, collection of disjoint tuples $(S_1, t_1), (S_2, t_2), ..., (S_k, t_k)$*
Question: *Find k mutually vertex-disjoint paths in G, connecting all $s_i \in S_i$ to t_i for each i, $1 \le i \le k$ s.t. each edge in E is not associated with more than c paths and (19) and/or (20) is minimum.*

Constraints (15), (16) and (21) still hold. Only constraint (14) is modified and redefined in (23). $|S_p|$ is the cardinality (i.e. the number of members) of S_p.

$$\begin{gathered} |S_p|x_{pij} - y_{phij-1} - y_{pvi-1j} - y_{phij} - y_{pvij} - q_{pi} = 0 \\ 1 \le p \le k \\ 1 \le i \le M \\ 1 \le j \le N \end{gathered} \tag{23}$$

5.2 Placement

Traditional placement and routing problems in VLSI CAD are considered as two separate problems. In this section, we present an ILP formulation where we solve both placement and routing of our sensor network simultaneously. In most cases, the medical application itself derives the location of sensors. However, in some applications, the flexibility of placing sensors within a specific range may be bestowed. This can be easily accommodated by our proposed ILP technique and the optimal location of sensor to minimize our objective function will be found. For instance, a temperature sensor may be required to be placed on chest, but its exact location is not given. In this case, our technique attempts to optimize its location too.

Problem: *Placement and Routing to Maximize the Fault Tolerance and Minimize the Power Consumption*
Instance: *Graph $G(V, E)$ s.t. G is a grid, fault occurrence weights on edges of G, collection of disjoint vertex set $(S_1, S'_1), (S_2, S'_2), ..., (S_k, S'_k)$* s.t. $s_i \in S_i \subseteq V$ and $t_i \in S'_i \subseteq V$, S_i and S'_i are set of nodes which can potentially be source or sink of a path.
Question: *Find k mutually vertex-disjoint paths in G, connecting s_i to t_i for every i, $1 \le i \le k$ s.t. (19) and/or (20) is minimum.*

In this problem, the nodes which are candidates to be either source or sink of the paths are given in

Proceedings of the 11th IEEE International Conference on Embedded and Real-Time Computing Systems and Applications (RTCSA'05)
1533-2306/05 $20.00 © 2005 IEEE

sets S_i's and S_i''s. For each tuple $(S_i, S'i)$, however, exactly one node from S_i and one node from S_i' will be selected. In the case where the position of a node is fixed (i.e. the node must be placed at a specific position), the set S_i or S_i' may have only one member. To solve this problem, our ILP formulation is modified and we define two sets of new variables for each communication pair.

$$\begin{array}{c} \sum_{s_{pij} \in S_p} s_{pij} = 1 \\ 1 \le p \le k \\ 1 \le i \le M \\ 1 \le j \le N \end{array} \quad (24)$$

$$\begin{array}{c} \sum_{s'_{pij} \in S'_p} s'_{pij} = 1 \\ 1 \le p \le k \\ 1 \le i \le M \\ 1 \le j \le N \end{array} \quad (25)$$

$$\begin{array}{c} 2x_{pij} - y_{phij-1} - y_{pvi-1j} - y_{phij} - y_{pvij} - s_{pij} - s'_{pij} = 0 \\ 1 \le p \le k \\ 1 \le i \le M \\ 1 \le j \le N \end{array} \quad (26)$$

The constraints (24) and (25) enforce that only one node from sets S_i or S_i' is selected as a source or a sink of a path. Constraint (26) is modified from (14) to ensure that a feasible path for every communication pair is found. Constraint (15), (16) and objective functions (19) and (20) still hold for this problem.
We can apply all the variations of routing problems described in Section 5.1 to the placement and routing formulations.

6 Experimental Analysis

In this Section, we present the simulation results on fault/power lifetime optimization on various benchmarks. We used the CPLEX solver on a 400MHz Sun Ultra-10 machine and allowed the maximal runtime to be 30 minutes. Various benchmarks are generated for the experimental analysis. The size of benchmarks varies from 5×5 to 11×11. The number of communication pairs are altered as well. The communication pairs are placed across the grid randomly for each instance size. The power consumption of the processing units is assumed to be $P_w = 24mW$ (Berkeley mica2dots - power supply voltage $\cong 3V$). We also assume that the system is equipped with ten $1.5V$, $2000mAh$ AA batteries. Hence, $\lambda_p = 0.0008h^{-1}$. Fault occurrence rate, λ_f, picks the values of $1/24h^{-1}$, $1/30h^{-1}$ and $1/36h^{-1}$. $\lambda_f = 1/24h^{-1}$ implies that on average, a random tear occurs in every 24 hours.

Table 1 illustrates the simulation results for a 5×5 grid with 8 communication pairs. The first and the second columns present the values of objective function (20) and (19), respectively. As shown in the Table 1, for small values of λ_f (i.e. $= 1/24$), the routing must be performed with most emphasis on power lifetime optimization while for larger λ_f's, (i.e. $= 1/36$), the fault lifetime is optimized to achieve the best results. For $\lambda_f = 1/30h^{-1}$, both fault and power objective functions yields approximately similar lifetimes. Table 2 exhibits similar simulation results for a 7×7 grid. Figure 7 represents fault/power lifetime ratio for a 7×7 grid. In this set of experiments, the location of source and sink nodes in each communication pair was selected randomly across the grid such that the rectilinear distance between source and sink nodes is not greater than l_{max} where $1 \le l_{max} \le 7$. Both the number of communication pairs and l_{max} are varied. Each plot in Figure 7 corresponds to a specific l_{max}. For each data-point, 20 benchmarks were generated and the average ratio of all 20 benchmarks is shown in the diagram. Overall, 1540 benchmarks were generated for $\lambda_f = 1/30h^{-1}$. Ratio greater than 1 indicates that the system is more sensitive to faults and most probably dies due to the faults while ratio less than 1 implies that most likely the system becomes paralyzed due to the power outage. According to Figure 7, the fault/power lifetime ratio drops to zero for large number of communication pairs distributed randomly with $l_{max} \ge 3$. Hence, more efforts must be spent on power lifetime optimization of such benchmarks while in less congested networks, fault lifetime gains more importance.

	Fault Lifetime (hours)	Power Lifetime (hours)
$\lambda_f = 1/24\,h^{-1}$	N/A	64.38
$\lambda_f = 1/30\,h^{-1}$	56.25	55.30
$\lambda_f = 1/36\,h^{-1}$	51.35	N/A

Table 1. Power/Fault Lifetime Comparison in 5x5 Grid

	Fault Lifetime (hours)	Power Lifetime (hours)
$\lambda_f = 1/24\,h^{-1}$	N/A	40.11
$\lambda_f = 1/30\,h^{-1}$	33.71	33.15
$\lambda_f = 1/36\,h^{-1}$	31.74	N/A

Table 2. Power/Fault Lifetime Comparison in 7x7 Grid

Figure 8 presents lower bounds for both power and fault lifetime of system. Each data-point again reflects the average of lifetime acquired from 20 benchmarks. In each benchmark, the source and the sink of communication pairs are placed with uniform distribution across the grid where l_{max} = instance size. As illustrated in the diagram, the solver was unable to find a

Proceedings of the 11th IEEE International Conference on Embedded and Real-Time Computing Systems and Applications (RTCSA'05)
1533-2306/05 $20.00 © 2005 IEEE

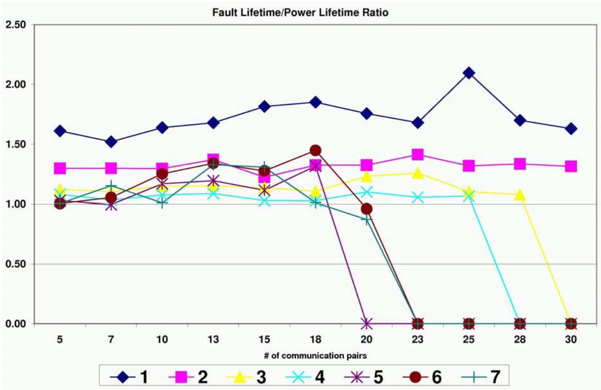

Figure 7. Fault/Power Lifetime Ratio for 7x7 Mesh Size

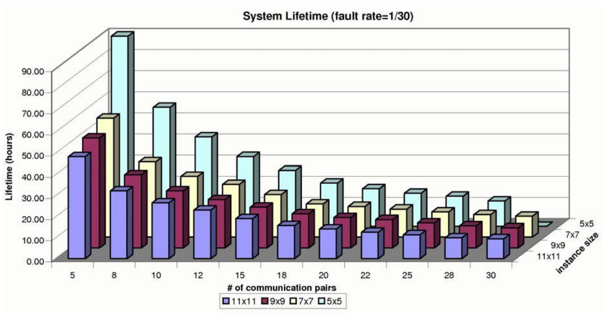

Figure 8. Lifetime Analysis ($\lambda_f = 1/30\, h^{-1}$)

feasible solution for any 30 communication pairs placed randomly on a 5×5 grid within 30 minutes.

In Figure 9, the lower bounds for both power and fault lifetime are represented where the size of the grid was set to 7×7 and l_{max} as well as the number communication pairs are varied accordingly. Each data-point is the average of 20 benchmarks. The final observation is that when the number of communication pairs grows, if the network becomes relatively congested, both power and fault lifetime become relatively constant and similar. The reason is that most resources of the system is already utilized and the routing algorithm do not have the flexibility of choosing the best solution with respect to either power optimized or fault tolerant routing techniques.

7 Conclusion

We proposed a new technique for placement and routing in sensors networks. Our technique employs ILP and generates a lower bound solution for both placement and routing problems. Our method accomplishes both placement and routing simultaneously and considers both fault tolerance and power optimization

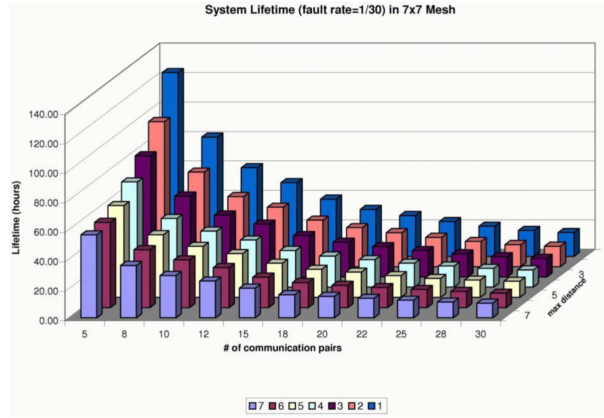

Figure 9. Lifetime Analysis on Fixed Mesh Size Varying Communication Distance($\lambda_f = 1/30\, h^{-1}$)

objectives. We showed that in order to improve the overall performance and lifetime of a sensor network, it may not be sufficient to focus on only one aspect of optimization and various factors may have to be considered.

References

[1] G. Barrenechea, B. Beferull-Lozano, and M. Vetterli. Lattice sensor networks: capacity limits, optimal routing and robustness to failures. In *IPSN'04: Proceedings of the third international symposium on Information processing in sensor networks*, pages 186–195. ACM Press, 2004.

[2] M. Bona. *Textile Quality*. Texilia, 1994.

[3] R. Jafari, F. Dabiri, P. Brisk, and M. Sarrafzadeh. Adaptive and fault tolerant medical vest for lifecritical medical monitoring. In *SAC'05: Proceedings of the 20th Annual ACM Symposium on Applied Computing*, March 2005.

[4] D. Meoli and T. May-Plumlee. Interactive electornic textile development: A review of technologies. *Journal of Textile and Apparel, Technology and Management*, 2(2), 2002.

[5] L. M. Ni and P. K. McKinley. A survey of wormhole routing techniques in direct networks. *IEEE Computer*, 26(2):62–76, 1993.

[6] A. Primentas. Puncture and tear of woven fabrics. *Journal of Textile and Apparel, Technology and Management*, 1(4), 2001.

[7] M. Sarrafzadeh and C. Wong. *An Intorduction to VLSI Physical Design*. McGraw-Hill, 1996.

[8] N. A. Sherwani. *Algorithms For VLSI Physical Design Automation*. Kluwer Academic Publishers, 1993.

[9] B. Witkowska and I. Frydrych. A comparative analysis of tear strength methods. *Fibers and Textiles in Eastern Europe*, 12(2), June 2004.

Proceedings of the 11th IEEE International Conference on Embedded and Real-Time Computing Systems and Applications (RTCSA'05)
1533-2306/05 $20.00 © 2005 **IEEE**