Defect Tolerance: Fundamental Limits and Examples

Jennifer Tang, Da Wang, Yury Polyanskiy, Gregory Wornell

***Abstract*—This paper addresses the question of how to add redundancy to a collection of physical objects so that the overall system is more robust to failures. Physical redundancy can (generally) only be achieved by employing copy/substitute procedures. This is fundamentally different from information redundancy, where a single parity check simultaneously protects a large number of data bits against a single erasure. We propose a bipartite graph model of designing defect-tolerant systems where defective objects are repaired by reconnecting them to strategically placed redundant objects. The fundamental limits of this model are characterized under various asymptotic settings and both asymptotic and finite-size optimal systems are constructed.**

Mathematically, we say that a k by m bipartite graph corrects t defects over alphabet of size q if for every q-coloring of k left vertices there exists a coloring of m right vertices such that every left vertex is connected to at least t same-colored right vertices. We study the tradeoff between redundancy m/k and the total number of edges in the graph divided by k. The question is trivial when $q \geq k$: the optimal solution is a simple t-fold replication. However, when $q < k$ some non-trivial savings are possible by leveraging the inherent repetition of colors.

***Index Terms*—defect-tolerant circuits, bipartite graphs, combinatorics, worst-case**

I. Introduction

Classical Shannon theory established principles of adding redundancy to data for combatting noise and, dually, of removing redundancy from data for more efficient storage. The central object of the classical theory is information, which unlike physical objects, can be freely copied and combined. In fact, the marvel of error-correcting codes is principally based on the counter-intuitive property that multiple unrelated information bits $X_1, \ldots, X_k$ can be simultaneously protected by adding "parity-checks" such as

$$Y = X_1 + \cdots + X_k \mod 2\,. \tag{1}$$

In this example, the added parity-check Y allows the recovery of the original message even if vector

$$(X_1, X_2, \ldots, X_k, Y)$$

undergoes an erasure of an arbitrary element.

Physical objects (e.g. transistors in a chip) may also be subject to erasures (failures) and thus it is natural to ask about ways of insuring the system against probable failure events. Necessarily, any such solution would entail addition of spare (redundant) elements. Note, however, that for physical objects operations such as (1) are meaningless: generally the only operations that apply to physical objects are copy/substitute. It may, therefore, seem that nothing better than a simple replication can guard against failures. This paper shows otherwise. Indeed, there exist non-trivial ways to add redundancy as long as the objects' diversity does not exceed their number. That is, if the number of types of objects is smaller than the total number of them.

Specifically, we study the following problem formulation: Given k objects ("functional nodes"), connect each one of them to some of the available m spares ("redundant nodes") in such a way that in the event that $t \geq 1$ of the objects fail (originals or spares) the overall system can be made to function after a repair step. Such a repair step consists of replacing each failed functional node with one of the spares that it is connected to. The key restrictions are 1) the functional nodes are one of q types 2) the spares have to be programmed to one of the q types *before* the failure events are known and 3) the same connections need to repair all possible choices of types for the k functional nodes. We are interested in minimizing the redundancy m/k and the number of connections to spare nodes.

Our motivation for studying this model comes from the following applications:

- Objects are digital gates of one of q types on a silicon chip. Imperfect manufacturing process causes certain gates to fail. As part of post-manufacture testing a configurable interconnect fabric is programmed to route around defective gates. Details are discussed further in Section II-B.
- Objects are elements in a programmable logic device (e.g. look-up tables (LUTs) in an FPGA). As part of periodical firmware update, manufacturer assigns values of LUTs (both functional and redundant) without knowledge of locations of device-specific failures. Then, a built-in algorithm for each failed LUT T reconnects it to an adjacent spare LUT R, with the requirement that R and T be equivalent. The key here is for the local algorithm to be computationally non-demanding.

It is not hard to come up with other potential applications in warehouse planning, operations research, public safety etc.

In short, we are looking for a $k \times m$ bipartite graph with the property that for any q-coloring of the left nodes there is a q-coloring of the right nodes such that each of the k nodes is connected to at least t nodes of its color. The goal is to trade off redundancy m/k vs. number of edges. For $q = 2$ our problem is equivalent to sparsity vs. edge-size tradeoff for (t, t)-colorable hypergraphs, cf. [1]. It may be instructive to

JT, YP and GW are with the Department of Electrical Engineering and Computer Science, MIT, Cambridge, MA 02139 USA. e-mail: {jstang,yp,gww}@mit.edu. DW was with the Department of Electrical Engineering and Computer Science, MIT, Cambridge, MA, 02139 and is now with Two Sigma Investments, New York, NY, 10013. e-mail: dawang@alum.mit.edu.

The research was supported by the Center for Science of Information (CSoI), and NSF Science and Technology Center, under grant agreement CCF-09-39370. This work was also supported in part by Systems on Nanoscale Information fabriCs (SONIC), an SRC STARnet Center sponsored by MARCO and DARPA.

978-1-5090-1805-5/16/$31.00 ©2016 IEEE

look at simple non-trivial graphs in Figures 3-4. (In all figures, circles are original or functional nodes and squares are spare or redundant nodes.)

To summarize our main findings, if the number of types $q \geq k$ then no strategy is better than straightforward t-fold replication. However, as long as $q < k$ there exists designs that provide savings compared to repetition, as we will see in Section III. Consequently, we characterize the fundamental tradeoff between redundancy m/k and the number of edges (connections) in the following cases: 1) q, t fixed and $k, m \to \infty$; 2) q fixed and $k, m, t \to \infty$; 3) q, k fixed and $m, t \to \infty$. Perhaps surprisingly, in this (combinatorial) problem it is possible to obtain exact answers for asymptotics.

II. Problem setup and main results

A. Defect-tolerance model

Definition 1. *Fix finite alphabet $\mathcal{X}$ where $|\mathcal{X}| = q$. A bipartite graph with k functional (left) nodes and m redundant (right) nodes is called a t-*error correcting design *if for any labeling of k functional nodes by elements of $\mathcal{X}$ there exists a labeling of m redundant nodes by elements of $\mathcal{X}$ such that every functional node labeled $x \in \mathcal{X}$ has at least t neighbors labeled x. We will call such a graph a $(k, m, t, E)_q$-design, with E denoting the number of edges.*

This paper is devoted to the fundamental tradeoff between the two basic parameters of t-error correcting designs:

- redundancy of a $(k, m, t, E)_q$-design is $\rho = \frac{m}{k}$
- the wiring complexity (or average degree) of a $(k, m, t, E)_q$-design is $\varepsilon = \frac{E}{k}$

Definition 2. *For a fixed q and $t \geq 1$ we define the region $\mathcal{R}_t$ as the closure of the set of all achievable pairs of (ε, ρ):*

$$\mathcal{R}_t \triangleq \text{closure}\left\{\left(\frac{E}{k}, \frac{m}{k}\right) : \exists (k, m, t, E)_q\text{-}design\right\} \qquad (2)$$

Proposition 1. *(Properties of $\mathcal{R}_t$)*

1) *$(\varepsilon, \rho) \in \mathcal{R}_t$ iff there exists a sequence of $(k, m, t, E)_q$-designs with $\frac{E}{k} \to \varepsilon$, $\frac{m}{k} \to \rho$ as $k, m \to \infty$;*
2) *if $(\varepsilon, \rho) \in \mathcal{R}_t$ and $\varepsilon' \geq \varepsilon, \rho' \geq \rho$ then $(\varepsilon', \rho') \in \mathcal{R}_t$;*
3) *$\mathcal{R}_t$ are closed convex subsets of $\mathbb{R}^2_+$;*
4) *We have*

$$\limsup_{t\to\infty} \frac{1}{t}\mathcal{R}_t = \text{closure}\left\{\bigcup_{t\geq 1} \frac{1}{t}\mathcal{R}_t\right\} \triangleq \mathcal{R}_\infty\,, \qquad (3)$$

5) *Limiting region $\mathcal{R}_\infty$ is also a closed convex subset of $\mathbb{R}^2_+$ characterized as*

$$\mathcal{R}_\infty \triangleq \text{closure}\left\{\left(\frac{E}{kt}, \frac{m}{kt}\right) : \exists (k, m, t, E) - design\right\} \qquad (4)$$

B. Reconfigurable defect-tolerant circuits

To interpret the relation between Definition 1 and defect-tolerance we consider one particular application, namely reconfigurable circuits. Consider a chip design process, in which the chip is composed of many similar cells (e.g. standard-cell designs of ASICs). Cell structure is dictated by the chip manufacturer (fab). Each cell has k input/output buses and k placeholders (nodes) that can be filled in with logic realizing one of q functions. Now because of manufacturing defects, not all k functional elements will operate correctly. For this reason, each cell also contains m placeholders for redundant elements. The designer then selects what type of logic to instantiate into these redundant elements. Once the chip is manufactured and placed on the testbed, the testing equipment goes over each cell and checks which functional elements came out defective. The programmable switches then can be used to reconnect input/output buses from the defective functional elements to one of the redundant nodes containing the same logic.

With respect to this application, our goal is to understand what cell topologies the fab should try to implement in order to attain optimal tradeoff between the number of redundant elements, provisional wires (buses) and defect-tolerance. The exact relation to the previous definition of the t-error correcting design is as follows: the k functional nodes in our model represents the placeholders intended for the components which are necessary for the chip to operate and the redundant nodes represents the placeholders for the redundant components. The labeling we apply to the nodes is the choice of components for each space. The edges correspond to the provisional wires.

Proposition 2. *An interconnect for a reconfigurable circuit can tolerate any t manufacturing defects for any choice of functional nodes if and only if the interconnect is a t-error correcting design.*

Note that our performance metrics, ρ and ε, are meaningful for this circuit interpretation: they correspond to the extra silicon area and wiring (and fan-out) required for defect-tolerance. There are certainly other metrics (such as geometric constraints) which are interesting for circuit applications, but we leave that to future work.

One may argue that the interconnect should be allowed to depend on the labeling of functional nodes. Indeed, the latter will be known before the final topology for the chip is made. In contrast, our procedure insists on laying out provisional wires before the specific choice of elements in the placeholders is known. The explanation is that our work attempts to find a universal design, which would be independent of the chosen functional node labels and thus could serve as the new standard cell for all defect-tolerant circuits. Nevertheless, we will discuss variations of this procedure in Section VI.

Relation to prior work: The subject of designing digital electronics robust to errors has been traditionally approached with the goal of combatting dynamic noise. This is epitomized in the large body of work started by von Neumann [2]. Although significant progress has been made in understanding fundamental limits in von Neumann's model, the practical applications are limited due to a prohibitively high level of redundancy required [3].

Instead, we are interested in fighting static manufacturing failures which has the advantage of being able to test which parts of the circuit failed and to configure out (or "wire around") the defective parts. This side information enables significant savings in redundancy [4] and it is rather popular in practice: multi-core CPUs [5], analog-to-digital converters [6], sense-amplifiers [7], parallel computing [8], [9], etc.

In summary, fighting dynamic noise (von Neumann's model) has good theoretical understanding, but requires huge redundancy. Static defects are practically handled via reconfigurability. This paper is an attempt to provide theoretical foundations for the latter method.

C. Overview of two main results

The two main results that characterize the redundancy-wiring complexity tradeoff are for the small t case (Theorems 3 and 4) and the asymptotic t case (Theorem 5).

Theorem 3. *When $q = 2$, for $t = 1$ and $t = 2$, we have*

$$\mathcal{R}_t = \{(\varepsilon, \rho) : \varepsilon \geq t \text{ and } \varepsilon \geq 2t - \rho\} \tag{5}$$

Theorem 4. *For $q = 3$ and $t = 1$ we have*

$$\mathcal{R}_1 = \{(\varepsilon, \rho) : \varepsilon \geq 1 \text{ and } \varepsilon \geq 3 - 2\rho\} \tag{6}$$

Theorem 5. *Fix q. The region $\mathcal{R}_\infty$ defined in* (3) *is the closure of the set of points $(\tilde{\varepsilon}, \tilde{\rho})$, parameterized by the distribution P_S with a finite support on $\mathbb{Z}_+$, and*

$$\tilde{\varepsilon} = \frac{\mathbb{E}[S]}{F(P_S)}, \quad \tilde{\rho} = \frac{1}{F(P_S)}, \tag{7}$$

$$F(P_S) \overset{\triangle}{=} \min_{P_X} \max_{P_{Y|\underline{L}}} \min_{j\in[q]} \frac{1}{P_X(j)} \mathbb{E}\left[L_j \mathbb{1}\{Y = j\}\right] \tag{8}$$

where $\mathbb{E}[\cdot]$ is computed over random variables $S \in \mathbb{Z}_+, X \in [q], \underline{L} = (L_1, \ldots, L_q) \in \mathbb{Z}_+^q, Y \in [q]$ with joint distribution

$$P_{S,\underline{L},Y}(s, \underline{\ell}, y) \overset{\triangle}{=} P_S(s) P_{\underline{L}|S}(\underline{\ell}|s) P_{Y|\underline{L}}(y|\underline{\ell}). \tag{9}$$

where

$$P_{\underline{L}|S}(\underline{\ell}|s) \overset{\triangle}{=} \binom{s}{\ell_1, \cdots, \ell_q} \prod_{j=1}^{q} P_X(j)^{\ell_j}. \tag{10}$$

Theorem 5 parametrically defines a region of achievable designs, whereas Theorem 3 and Theorem 4 explicitly define the respective achieveable regions. (Note that evaluation of the bound (7) presents non-trivial technical difficulties.) The resulting achievable regions for $q = 2$ are plotted in Figure 1. This plot, for instance, shows that at redundancy level 10% we can:

- correct 1 error if each functional node is connected on average to about 1.9 redundant nodes
- correct 2 errors if each functional node is connected on average to about 1.9×2 redundant nodes
- correct 10^3 errors if each functional node is connected on average to about 1.7×10^3 redundant nodes

The optimal designs for Theorem 5 are what we call *subset designs*, which are discussed in Section III-B. The proofs of Theorems 3 and 5 are given in Sections IV and V respectively. See Section IV of [10] for Theorem 4.

D. Implications and extensions of results

The result for $\mathcal{R}_1$ and $\mathcal{R}_2$ demonstrates that for correcting small numbers of defects the best solution in the limit of a large number of functional nodes is a linear combination of two basic designs, the repetition block and the complete design (see Figure 2), and designs with finite k can do no better.

We note that while we do not know $\mathcal{R}_t$ for $t > 2$, according to (4) all regions $\frac{1}{t}\mathcal{R}_t$ will lie between $\mathcal{R}_1$ and $\mathcal{R}_\infty$, approaching the latter as $t \to \infty$, making Theorem 5 the fundamental limit for the tradeoff between redundancy and wire complexity. It is perhaps surprising that unlike most known asymptotic combinatorial problems, this one admits a relatively simple solution. Theorem 5 also holds for arbitrary alphabet, whereas computing regions $\mathcal{R}_t$ for large q is not covered by Theorem 3 or 4.

We also study asymptotics in the regime of fixed k and $m, t \to \infty$ (see [10] Section V).

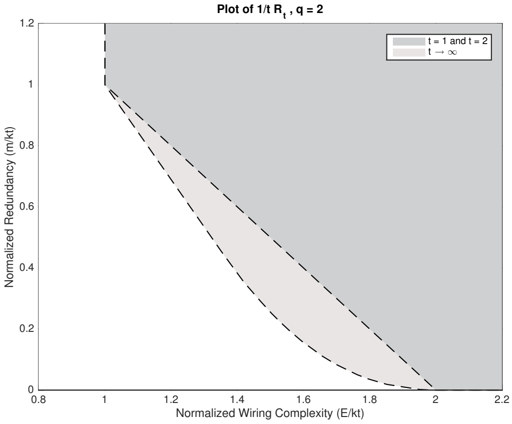
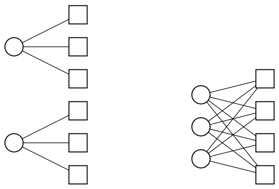


Fig. 1. Achievable regions for redundancy and wiring complexity tradeoff when $q = 2$. Regions $\mathcal{R}_1$ and $\frac{1}{2}\mathcal{R}_2$ are shown in darker gray. Region $\mathcal{R}_\infty$ includes lighter and darker gray areas. All other regions $\frac{1}{t}\mathcal{R}_t$ lie between $\mathcal{R}_1$ and $\mathcal{R}_\infty$.



(a) Repetition block (b) Complete design

Fig. 2. Two elementary designs.

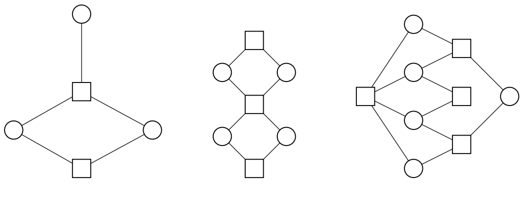
III. Examples of good designs

The two most basic designs are the following:

1) *Repetition blocks:* Each functional node has t private redundant nodes. (See Figure 2(a).) Corrects t errors.
2) *Complete designs:* Fully connected bipartite graph with qt redundant nodes. (See Figure 2(b).) Corrects t errors (label redundant nodes with t copies of each value in $\mathcal{X}$).

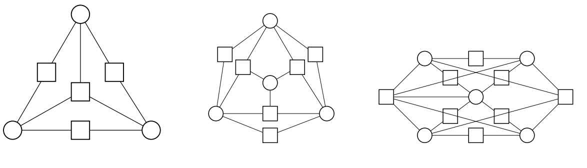
A. Smallest non-trivial designs

If $k \leq q$ then all functional nodes can have different values and thus one is forced to use the repetition block to correct t errors. For $k = q + 1$ the question becomes more interesting. First, notice that the minimal possible m equals qt, like in the complete design. However, some of the edges can be removed. Optimal designs with $k = q + 1$, $m = q$ and $t = 1$ are shown in Figure 3. Optimal designs with $k = q + 1$, $m = 2q$ and $t = 2$ are shown in Figure 4. While these designs are optimal for a given value of k, we can find other designs with larger values of k which are better in terms of the ρ-ε tradeoff.



(a) $q = 2$ (b) $q = 3$ (c) $q = 4$

Fig. 3. Smallest non-trivial 1-error correcting designs.



(a) $q = 2$ (b) $q = 3$ (c) $q = 4$

Fig. 4. Smallest non-trivial 2-error correcting designs.

B. Subset designs

$S(k, s)$ is a bipartite graph with k functional nodes and $m = \binom{k}{s}$ redundant nodes, each connected to a distinct s-subset of $\{1, \ldots, k\}$. In general, we allow subset designs to have multiple and possibly different subset sizes. We need the following idea to make this precise.

Proposition 6 (Merging). *Consider $(k, m_j, t_j, E_j)_q$-designs G_j. The merging of G_j, denoted $G = \bigvee_j G_j$ is a graph formed by taking disjoint copies of G_j and identifying functional nodes. G is a $(k, \sum_j m_j, \sum_j t_j, \sum_j E_j)_q$-design.*

Definition 3 (Subset design). *Given k and positive integers $s_1, s_2, \ldots, s_r \in [k]$, $S(k, s_1) \vee S(k, s_2) \vee \cdots \vee S(k, s_r)$ is a subset design with k functional nodes and $m = \sum_{j=1}^{r} \binom{k}{s_j}$ redundant nodes.*

For example, the Hamming block, Figure 4(a), is $S(3, 2) \vee S(3, 3)$, the repetition block is $S(k, 1) \vee \cdots \vee S(k, 1)$ (t times) and the complete design is $S(k, k) \vee \cdots \vee S(k, k)$ (qt times).

Subset designs are the unique designs where there exists a group of bipartite-graph automorphisms that acts as the full symmetric group S_k on functional nodes. The next proposition gives an estimate on the performance of subset designs. (See [10] Section V for exact statement.)

Proposition 7. *(Informal) Fix $\mathcal{X}$, $q = |\mathcal{X}|$ and $k \in \mathbb{Z}$. If G is a subset graph with proportion of degree s redundant nodes given by P_S, k functional nodes, and m redundant nodes, then G has $E = m\mathbb{E}[S]$ edges and G can correct*

$$t \approx \frac{m}{k} F(P_S) \tag{11}$$

errors.

Proof Sketch. To show $t \approx \frac{m}{k} F(P_S)$, fix any labeling $w^k \in \mathcal{X}^k$ of the k functional nodes of G. Let r^m to be the optimal labeling of the redundant nodes. Let

- P_X denote the empirical distribution of the frequency of each label in w^k
- $\underline{\ell} = (\ell_1, \cdots, \ell_q)$ be the *type* of each redundant node v, where ℓ_j is the number of functional nodes with label j which is a neighbor of redundant node v
- $P_{Y|\underline{L}}(j|\underline{\ell})$ be the proportion (empirical distribution) of redundant nodes of type $\underline{\ell}$ which are labeled j in labeling r^m

The distribution of $\underline{\ell}$ for degree s redundant nodes is approximately given by (10). For each label j, we can count the *average* number of redundant node neighbors with label j a functional node with label j has. This quantity is given by (8) without the maximums and minimums, which we get after taking the worse case label j and P_X with the best possible $P_{Y|\underline{L}}(j|\underline{\ell})$. We can show there is a way for r^m to obtain this average for each functional node by random coding.

□

IV. Bounds for finite t

A. Elementary achievability

Proposition 8. *The following region is achievable for any $t \geq 1$ and $q \geq 2$:*

$$\mathcal{R}_t^{(K)} \triangleq \{(\varepsilon, \rho) : \varepsilon \geq qt + (1 - q)\rho, \varepsilon \geq t, \rho \geq 0\} \tag{12}$$

Proof. Note that corner points (t, t) and $(qt, 0)$ are achieved by the repetition block and the complete design, respectively. By Proposition 1 the region $\mathcal{R}_t$ is convex and hence must contain $\mathcal{R}_t^{(K)}$. □

By merging the repetition block and complete design, we can get designs in $\mathcal{R}_t^{(K)}$ with each functional node having the same degree.

B. Covering converse

Theorem 9. *Fix $\mathcal{X}$, $q = |\mathcal{X}|$ and suppose $(\varepsilon, \rho) \in \mathcal{R}_t$. Then there exists $\pi_t, \pi_{t+1}, \ldots, \pi_{qt} \geq 0$ satisfying*

$$\sum_{j=t}^{qt} j\pi_j \leq \varepsilon, \quad \sum_{j=t}^{qt} \pi_j = 1 \tag{13}$$

$$\sum_{j=t+1}^{qt} \pi_j \log_q \lfloor j/t \rfloor \geq 1 + (t - 1)\pi_t - \rho\,. \tag{14}$$

Proof. Notice that every functional node clearly should have degree at least t. Let us define $\pi_j, j = t, t + 1, ..., qt - 1$ to be the fraction of functional nodes of degree j and π_{qt} to be the fraction with degree qt or larger. This satisfies (13). We only need to show (14).

For each labeling $r^m \in \mathcal{X}^m$ of redundant nodes let $\mathcal{G}_t(r^m)$ be the set of functional node labelings for which conditions of Definition 1 are satisfied (we say that r^m covers $\mathcal{G}_t(r^m)$ of the labelings). It is clear that the design is t-error correcting if and only if

$$\left| \bigcup_{r^m \in \mathcal{X}^m} \mathcal{G}_t(r^m) \right| = |\mathcal{X}|^k = q^k\,. \tag{15}$$

Two functional nodes of degree t should have disjoint neighborhoods (otherwise labeling them different values clearly violates Definition 1). Thus $\mathcal{G}_t(r^m)$ is empty unless each such neighborhood has a constant label. This shows that for the $tk\pi_t$ redundant nodes we are restricted to only $q^{k\pi_t}$ choices, while the rest contribute $q^{m-tk\pi_t}$ more choices.

Given any of the $q^{m-(t-1)k\pi_t}$ choices of r^m we can estimate $|\mathcal{G}_t(r^m)|$ from above by assuming that each functional node of degree d can take any of the $\lfloor d/t \rfloor$ values in $\mathcal{X}$ while still satisfying the t-wise coverage condition of Definition 1. This yields

$$|\mathcal{G}_t(r^m)| \leq \prod_{j=t}^{qt} \lfloor j/t \rfloor^{k\pi_j}\,, \tag{16}$$

and thus applying the union bound to (15), we get (14). □

Proof of Theorem 3. Achievability follows from Proposition 8. The converse is given by evaluating the optimal ε for each ρ with the constraints given in (13)-(14) for $t = 1, 2$. □

Remark 1. *The bound given by Theorem 9 is the best bound of $\mathcal{R}_t$ known to us for values of ε near qt.*

V. Fundamental limit for $t \to \infty$

A. Converse and proof outline

Proposition 10 (Symmetrization Converse)**.** *If there exists a $(k, m, t, E)_q$-design then there exists a $(k, m \cdot k!, t \cdot k!, E \cdot k!)_q$-design which is a subset design (Definition 3).*

Proof. Let G be a $(k, m, t, E)_q$-design. Choose an ordering of the functional nodes in G. For each $\sigma \in S_k$ (the full symmetric group on k elements), let G_σ be isomorphic to the design G, with the order of its functional nodes transformed by σ. Then merge G_σ for all $\sigma \in S_k$ identifying functional nodes with the same order, so that the result is

$$G_{\mathrm{PERM}} = \bigvee_{\sigma \in S_k} G_\sigma \,. \tag{17}$$

G_{PERM} is constructed to be permutation invariant (and thus a subset design) and by Proposition 6 G_{PERM} is a $(k, m \cdot k!, t \cdot k!, E \cdot k!)_q$-design. □

Proof Sketch. Proof of Theorem 5

Achievability: Given by Proposition 7. *Converse*: For any design G which is a $(k, m, t, E)_q$-design, there exists a subset design G' which is a $(k, m \cdot k!, t \cdot k!, E \cdot k!)_q$-design by Proposition 10. If P_S are the proportions of degree s redundant nodes in G', then by Proposition 7, we get $t \cdot k! \leq \frac{m \cdot k!}{k} F(P_S)$, bounding the performance of G. □

VI. Discussion

A. Comparison to other models for defect-tolerance

This paper studies a defect-tolerance model where steps proceed as follows:

a. bipartite graph is designed;
b. functional nodes get q-ary labeling;
c. redundant nodes are assigned q-ary labels (so that each functional node has t neighbors with matching label).

There are two natural variations where sequence of steps are interchanged:

- *adaptive graph*: b.→a.→c.
- *non-adaptive redundancy*: a.→c.→b.

In the first case, the graph is a function of the q-ary labels, while in the second case the redundant nodes are not allowed to depend on the labeling of functional nodes. The setting considered in this paper (a.→b.→c.) is an intermediate case.

The fundamental redundancy-wiring complexity tradeoff for these cases is defined similarly to (2). Both tradeoffs are rather easy to determine for any $t \geq 1$:

- *adaptive graph*: $\mathcal{R}_t = \{(\varepsilon, \rho) : \varepsilon \geq t, \rho \geq 0\}$.
- *non-adaptive redundancy*: $\mathcal{R}_t = \{(\varepsilon, \rho) : \varepsilon \geq qt, \rho \geq 0\}$.

These observations are summarized in Figure 5.

B. Stochastic defects

This work considers correcting arbitrary (worst-case) defect patterns. One of the conclusions is that to correct fraction α of defects (i.e. $t = \alpha k$) on k functional nodes, the number of edges should grow as k^2. Instead we can relaxed the requirement to correcting i.i.d. Bernoulli(α) defects. Each defect pattern will occur with some probability and we only want all defects in the design to be corrected with high probability (computed over distribution of defects and functional labels). It turns out that in such a probabilistic model, correcting fraction-α of defects is possible with designs possessing $O(k \log k)$ edges and $O(k)$ redundant nodes. See Section 4.4 in [11].

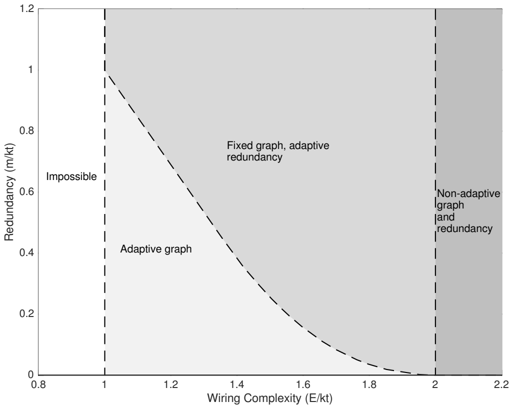


Fig. 5. Comparison of redundancy-wiring complexity tradeoffs for different levels of adaptivity when $q = 2$.

C. Open Problems

Regions which are still to be determined include:

- $\mathcal{R}_t$ for $t > 3$ and $q = 2$
- $\mathcal{R}_t$ for $t > 1$ and $q \geq 3$

For $q = 2$, it is also unknown what the smallest value of t is for which $\mathcal{R}_t$ does not equal the region defined in Equation (5).

References

[1] N. Alon and U. Feige, "On the power of two, three and four probes," in *Proc. 20th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*. Society for Industrial and Applied Mathematics, 2009, pp. 346–354.

[2] J. von Neumann, "Probabilistic logics and the synthesis of reliable organisms from unreliable components," *Automata studies*, vol. 34, pp. 43–98, 1956.

[3] G. Norman, D. Parker, and M. Kwiatkowska, "Evaluation the reliability of defect-tolerant architectures for nanotechnology with probabilistic model checking," in *Proc. 17th International Conference on VLSI Design*, 2004, pp. 907–912.

[4] K. Nikolic, A. Sadek, and M. Forshaw, "Fault-tolerant techniques for nanocomputers," *Nanotechnology*, vol. 13, no. 3, p. 357, 2002.

[5] D. Gizopoulos, M. Psarakis, S. V. Adve, P. Ramachandran, S. K. S. Hari, D. J. Sorin, A. Meixner, A. Biswas, and X. Vera, "Architectures for online error detection and recovery in multicore processors," in *Design, Automation Test in Europe Conference Exhibition (DATE)*, 2011, pp. 1–6.

[6] M. Flynn, C. Donovan, and L. Sattler, "Digital calibration incorporating redundancy of flash adcs," *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing*, vol. 50, no. 5, pp. 205–213, May 2003.

[7] N. Verma and A. Chandrakasan, "A 256 kb 65 nm 8t subthreshold sram employing sense-amplifier redundancy," *IEEE Journal of Solid-State Circuits*, vol. 43, no. 1, pp. 141–149, 2008.

[8] T. Leighton and C. Leiserson, "Wafer-scale integration of systolic arrays," *IEEE Transactions on Computers*, vol. C-34, no. 5, pp. 448–461, May 1985.

[9] J. Heath, G. Kuekes, and R. Williams, "A defect tolerant computer architecture: Opportunities for nanotechnology," *Science*, vol. 80, pp. 1716–1721, 1998.

[10] J. Tang, D. Wang, Y. Polyanskiy, and G. Wornell, "Defect tolerance: Fundamental limits and examples," *Preprint*, Jan. 2016, http://people.lids.mit.edu/yp/homepage/data/defects.pdf.

[11] D. Wang, "Computing with unreliable resources: Design, analysis and algorithms," Ph.D. dissertation, Massachusetts Institute of Technology, June 2014.