Stochastic Nanoscale Addressing for Logic

Eric Rachlin and John E. Savage
Department of Computer Science
Brown University

Abstract—**In this paper we explore the area overhead associated with the stochastic assembly of nanoscale logic. In nanoscale architectures, stochastically assembled nanowire decoders have been proposed as a way of addressing many individual nanowires using as few photolithographically produced mesoscale wires as possible. Previous work has bounded the area of stochastically assembled nanowire decoders for controlling nanowire crossbar-based memories. We extend this analysis to nanowire crossbar-based logic and bound the area required to supply inputs to a nanoscale circuit via mesoscale wires. We also relate our analysis to the area required for stochastically assembled signal-restoration layers within nanowire crossbar-based logic.**

I. Introduction

Before computer chips can be assembled with nanowires (NWs) (i.e $<$ 5nm feature sizes), a number of key manufacturing and design challenges must be addressed. These include developing area-efficient methods for controlling NWs using a limited number of photolithographically produced mesoscale wires (MWs). The interface between NWs and MWs is called a **NW decoder**. NW decoders for memories have been proposed for nanoscale memories consisting of NW crossbars. The goal is to address one (or a few) NW(s) in each of the crossbar's two dimensions. This allows a bit of data to be written by altering the physical state of the nanowires' point(s) of intersection (see Figure 1). A range of technologies have been presented to realize decoders for memories including using masks [1], [2], axial NW encoding [3], [4], radial NW encoding [5], random particle deposition [6], rotational offsets of intersecting sets of wires [7], [8], and micro-to-nano addressing blocks [9].

Each of these memory decoder technologies introduces randomness into the assembly process. Each MW controls (i.e. "turns off") some subset of the NWs in one dimension of a crossbar, but whether a given NW is controlled by a given MW is probabilistic, not deterministic. The probabilities in question are a function of the decoders' method of assembly. We have analyzed the area occupied by memory decoders and crossbars for a variety of stochastic assembly methods to determine the conditions under which the area is minimized. We have studied encoded NW decoders [10], randomized-contact decoders [6], masked-based decoders [11], and decoders for radially encoded NWs [5]. In all cases tight bounds have been derived on M, the number of MWs required to address N_a out of N NWs with high probability. From M, N_a and N, the area of a decoder and crossbar-based memory can be bounded.

In this paper we treat a related problem, namely, bounding the area of **decoders for logic circuits**. Let such a decoder have M MWs and N NWs. For a given method of stochastic decoder assembly, the goal is to choose M and N so there exists N_a NWs such that each of the 2^{N_a} subsets of the N_a NWs can be addressed by activating some subset of the MWs. If this holds, the N_a NWs can serve as inputs to a circuit.

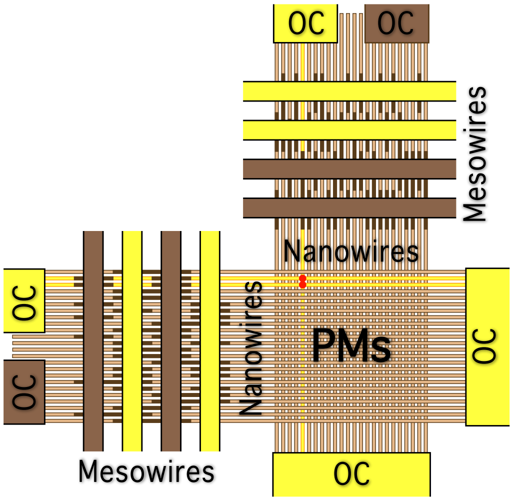

Fig. 1. A crossbar formed from two orthogonal sets of NWs in which programmable molecules (PMs) are located at the NW crosspoints. The NWs along each dimension are divided into groups by connecting them to ohmic contacts (OCs). To address a NW in one dimension, an OC is activated and mesoscale wires (MWs) are used to turn off all but NW address within that group. Each MW provides control over a subset of NWs, but these subsets are determined by a stochastic assembly process. The stochastically assembled coupling of MWs to NWs is referred to as a NW decoder. By addressing NWs along each dimension of the crossbar, the NW decoders provide control over NW crosspoints. Data is stored to an addressed crosspoint by application of a sufficiently large electric field. Data is sensed with a smaller field. In this figure, the same bit of information is stored at two crosspoints.

If the N input NWs form a crossbar with a second orthogonal set of NWs, the junctions between these two sets can be programmed. Each NW in the second set forms a WIRED-OR with the NWs to which it is connected (see Figure 2) [12]. When these "gates" are combined with inverters, a complete basis for nanoscale logic circuits is achieved. Also, the configuration of the WIRED-OR gates allows the $N - N_A$ unused input NWs to be disconnected from the logic circuit.

A. Paper Overview

In this paper we explore the overhead associated with stochastic assembly of crossbar-based logic decoders. We also examine the impact of manufacturing errors, in which some MWs only partially control some NWs. Sections II and III

This work was supported in part by NSF Grant CCF-0726794.

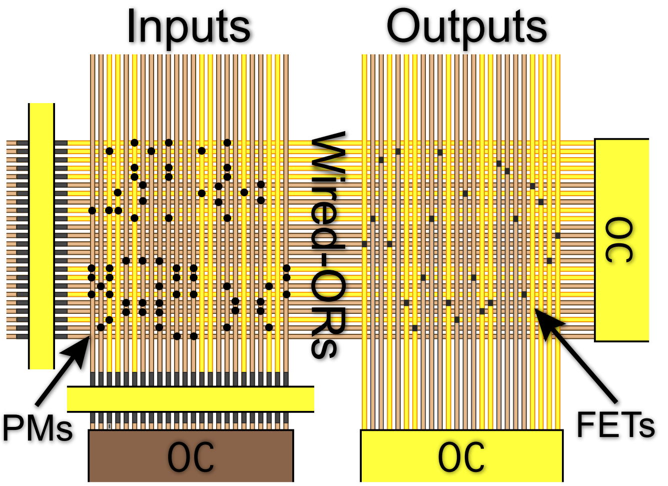

Fig. 2. A level of reconfigurable crossbar-based logic in which a WIRED-OR operation is followed by a signal restoration operation that also implements negation. Light NWs indicate that a boolean value of "1" is being applied, dark NWs indicate a "0". The two operations collectively implement a WIRED-NOR, and thus form a complete basis for boolean logic. The WIRED-OR operation is implemented like a read operation, except that multiple vertical NWs, and all horizontal NWs, are addressed. Any horizontal NW which is connected to an addressed input NW carries a current. The current carrying horizontal NWs then gate (i.e. make nonconducting) a subset of the output NWs using field-effect transistors (FETs). The diode connections used to perform the WIRED-OR operation can be configured via write operations using a NW decoder (not shown), which is disconnected during normal operation. The FETs used to implement the restoration operation may be placed stochastically.

model several types of NW decoder, highlighting the stochastic aspects of their assembly process, and explicitly defining requirements the decoders must meet to control memories and circuits. Section IV demonstrates that stochastically assembled NW decoders can supply inputs to crossbar-based logic using only small constant factor overhead. Section V presents an information-theoretic lower bound on this overhead. Finally, Section VI relates the analysis of the previous two sections to proposed stochastically-assembled inversion (and buffering) layers within nanoscale logic.

II. Modeling Nanowire Decoders

In a NW memory decoder in which M MWs are used to address N_A out of N NWs, each MW provides control over a random subset of NWs. Which NWs a given MW controls (makes nonconducting) is determined by the decoder's stochastic assembly process. A range of proposed decoders can be modeled using a **binary model with errors** [6].

For each NW, $\boldsymbol{n_i}$, we describe the subset of MWs that control it using a binary M-tuple, $\boldsymbol{c^i}$, called its **codeword**. A 1 in the j^{th} position, $\boldsymbol{c^i_j}$, indicates that the j^{th} MW, $\boldsymbol{m_j}$, controls $\boldsymbol{n_i}$. A 0 indicates that $\boldsymbol{m_j}$ is noncontrolling. An e in the j^{th} position indicates an error. In this case the j^{th} MW provides only partial control over a NW. Even in the presence of errors, we can conservatively assert that NW $\boldsymbol{n_i}$ is **reliably off** if some MW, $\boldsymbol{m_j}$, for which $\boldsymbol{c^i_j} = 1$, is turned on. Similarly, $\boldsymbol{n_i}$ is **reliably on** if only MWs for which $\boldsymbol{c^i_j} = 0$ are on.

The value of this binary decoder model with errors is that it allows us to succinctly establish criteria that NWs must satisfy to be addressable, while still accounting for the possibility of manufacturing errors. If a NW is reliably on while all other NWs are reliably off, it is **individually addressed** even when some MW/NW junctions contain errors. This model also lets us describe a decoder's assembly processes in terms of the probability distribution with which codewords are assigned.

A **randomized-contact decoder** (RCD) is any decoder that is assembled such that each bit of each codeword can be modeled as an independent random variable [13], [6]. Here $\boldsymbol{c^i_j} = 1$ with probability p, $\boldsymbol{c^i_j} = 0$ with probability q, and $\boldsymbol{c^i_j} = e$ with probability $r = 1 - (p + q)$. In an error-free **encoded NW decoder**, codewords are all drawn from some set, $\mathcal{C} \subset \{0,1\}^M$ [10]. During decoder assembly each NW codeword, $\boldsymbol{c^i}$, is modeled as a uniformly distributed independent random variable. One possible choice for $\mathcal{C}$ is an (h, M)**-hot code** in which codewords contains exactly h 1's and $M-h$ 0's. This ensures that a NW with a unique codeword is individually addressable. In an encoded NW decoder, errors may occur within a NW's codeword due to axial misalignment [5]. Rather than modeling the distribution of errors within codewords, we use p_f to denote the probability that a NW is misaligned, then assume that misaligned NWs are not used.

III. Requirements for Memory and Logic Decoders

We now consider the requirements memory and logic decoders must satisfy. Consider first **memory decoders**. Given N NWs along each dimension of a crossbar memory, we wish to construct a decoder such that N_A disjoint sets of NWs are individually addressable with probability $1 - \epsilon$. This ensures that data in a crossbar can be stored at N_A^2 distinct memory locations. A similar, but slightly stricter requirement is for N_A NWs along each dimension to be individually addressable. In either case, the NW decoder can be considered efficient if the number of MWs, M, is close to $\log_2 N_A$ and N_A is close to N. Both encoded NW decoders and RCDs have been shown to be efficient [10] [6] in that $\log_2 N_A/\epsilon < M < 3\log_2 N_A/\epsilon$ when N_A is a large fraction of N.

Now consider **logic decoders** that have N input NWs to a crossbar-based logic circuit. We desire a logic decoder with a set of $N_A < N$ NWs such that all 2^{N_A} subsets of these NWs can be addressed by applying inputs to M MWs. This ensures that all 2^{N_A} possible binary inputs can be supplied to the circuit. A seemingly alternative condition is that some set of N_A MWs exists such that these MWs are capable of addressing 2^{N_A} different subsets of NWs. The following lemma shows that these conditions are identical.

If a decoder is able to address all 2^{N_A} subsets of a set of N_A NW, the set of NWs is said to be **fully addressable**. Given a set of N_A NWs and a set of N_A MWs, we say the sets are **uniquely coupled** if each of the N_A NWs is controlled by a unique MW. In other words, each of the N_A MWs provides individual control over a distinct NW. This provides us with the criteria a logic decoder must satisfy.

Lemma III.1 *In a simple NW decoder, a set of N_A NWs is fully addressable if and only if their exists a set of N_A MWs to which it is uniquely coupled.*

Proof: Let S be the fully addressable set of N_A NWs. For each NW $\boldsymbol{n_i}$, consider the set $S_i = S - \boldsymbol{n_i}$. Since S_i is addressable, there must be a MW that uniquely controls $\boldsymbol{n_i}$. For the other direction, simply note that any set of NWs can be addressed by activating the MWs that uniquely control the NWs not in the set. ■

In order to bound the area required for stochastically assembled NW logic decoders, we wish to bound the number of MWs, M, and NWs, N, such that there exists a set of N_A fully addressable NWs with probability at least $1 - \epsilon$. By the above lemma, the probability that such a set exists is equal to the probability that there exists N_A NWs and N_A MWs such that each of the NWs is controlled by a unique MW. Furthermore, this condition is sufficient even for decoders that contain errors (i.e. MWs that only partially control certain NWs).

For a logic decoder to be considered efficient, M should be proportional to N_A. We note also note that is is possible to supply N_A inputs to a circuit without having a fully addressable set of N_A NWs if inputs are fed in sequentially, using a nanoscale memory as a buffer. This would be significantly slower, as N_A write operations would be required per input, but it would replace the need for a logic decoder. Instead a memory decoder would suffice.

A. Simple Versus Compound Decoders

Before proceeding with our area analysis, we consider the use of simple versus compound NW decoders. For memory decoders, compound decoders have been shown to greatly reduce the required number of MWs. Memory decoders are far more efficient if g groups of N NWs are each connected to separate OCs [10], [6]. This is not the case for logic decoders.

First notice that if a set of N_A NWs are uniquely coupled to N_A MWs, the N_A NWs are fully addressable whether or not they are all connected to a single OC. Now consider a compound NW decoder with g OCs, $N = gw$ total NWs and a set, S, of N_A fully addressable NWs. If NWs $\boldsymbol{n_i}, \boldsymbol{n_j} \in S$ are both connected to the same OC, then for the decoder to address the set $S_i = S - \boldsymbol{n_i}$ there must be a MW that is uniquely coupled to $\boldsymbol{n_i}$. If most NWs in S share an OC with at least one other NW in S, there must exist close to N_A uniquely coupled NWs and MWs. Thus, if the g OCs were replaced with a single larger OC, the resulting simple decoder would still have close to N_A fully addressable NWs.

Finally, notice that by replacing g OCs with a single OC, the added space between the g OCs can be eliminated. If additional NWs are then added such that N_A (as opposed to close to N_A) fully addressable NWs exist with probability at least $1 - \epsilon$, we can expect the resulting simple decoder to have close to the same area as the compound NW decoder it replaced. For this reason, the remainder of this paper is focused on bounding the area of simple NW decoders.

B. Post Assembly Configuration

In a NW decoder for logic with a set of N_A fully addressable NWs, the remaining $N - N_A$ NWs can be disconnected from all perpendicular NWs, and thus safely ignored. This post-assembly configuration is accomplished by using a memory decoder to program the WIRED-OR portion of the nanoscale circuit (see Figure 2) [12]. Arguably, one might consider decoding technologies by which the logic decoder itself could be constructed via similar post-assembly configuration [14]. In this scenario, stochastically assembled memory decoders would be constructed to individually address N_A input NWs, then write operations would permanently couple each of the NWs to a different MW. If this approach proves feasible, it would be quite efficient. The resulting logic decoder would consist of exactly N_A NWs and N_A MWs. The only additional overhead required is that of the the memory decoder. In the remainder of this chapter, we assume that connections between MWs and NWs are not programmable.

IV. Stochastic Assembly of NW Logic Decoders

One approach for producing a set of N_A fully addressable NWs is to connect N_A OCs to groups of w NWs, then select only one NW from each group. This is the deterministic decoder proposed by DeHon in [12], in which each of M MWs is coupled to a block of w NWs. The $M = N_A$ MWs fully address N_A out of $N = 2wN_A$ NWs. Here the factor of 2 approximates the space between the groups of w NWs. Depending on what is attainable through photolithography, the factor may be closer to 1. Assuming MWs are perpendicular to NWs, this decoder uses area

$$A \approx (2wM\lambda)(Nw\lambda) \approx 4w^2 N_A^2 \lambda^2$$

where λ and $2w\lambda$ is the pitch of MWs and NWs, respectively. We show that stochastically assembled decoders use less area.

A. Area Bounds

$2wMN\lambda^2$ is the area of a NW decoder with N NWs and M perpendicular MWs. To minimize this area, we must minimize MN while choosing M and N such that their exists a set of N_A fully addressable NWs with probability at least $1 - \epsilon$. To outperform the deterministic construction above we must have $MN < wN_A^2$ (or $2wN_A^2$ assuming w NWs between each OC).

To bound MN, one option is to consider the two extreme cases in which either $N = N_A$, or $M = N_A$. As shown in [15] however, these do not represent efficient solutions. For both encoded NW decoders and RCDs, when $N = N_A$, $M = O(N_A \ln N_A/\epsilon)$, and when $M = N_A$, $N = O(N_A \ln N_A/\epsilon)$. Since $MN = O(N_A^2 \ln N_A/\epsilon)$, setting $N = N_A$ or $M = N_A$ can only outperform the deterministic construction when N_A is small, i.e. $\ln N_A/\epsilon < w$.

For intuition behind this result, consider an encoded NW decoder in which $(1, M)$-hot codes are used. When $M = N_A$, in the absence of misalignment errors, each NW is controlled by exactly one MW. Furthermore, each NW is equally likely to be controlled by any given NW, so we must ask how many NWs are needed such that, with probability $1 - \epsilon$, each MW controls at least one NW. This is analogous to the classic **Coupon Collectors Problem** in which one of C coupons is selected with equal probability during each of T trials. The number of trials before all C coupons have been collected with probability $1-\epsilon$ is $T \approx C \ln C/\epsilon$. Thus in the error-free encoded NW decoder, $N \approx N_A \ln N_A/\epsilon$. Furthermore, if misalignment errors cause trials (NWs) to fail to collect any coupon with probability p_f, $N \approx (N_A/(1-p_f)) \ln N_A/\epsilon$ [11].

In order to obtain a more area efficient decoder, we can set $N = M = \beta N_A$ for some relatively small value of β. First, consider an encoded NW decoder in which $(1, M)$-hot codes are used. In the absence of misalignment errors, each NW is controlled by exactly one MW. Here we can again view MWs as coupons and NWs as trials. We require that C/β of the $C = M = \beta N_A$ coupons be collected over $T = C$ trials. Now let t_i be the number of trials required to collect the i^{th} coupon after the $(i-1)^{th}$ coupon has been collected (so $t_1 = 1$). When $i-1$ coupons have been collected, the probability that a trial collects a new coupon is $(C-i+1)/C$, so $E[t_i] = C/(C-i+1)$.

The number of trials required to collect κC coupons is $T_\kappa = \sum_{i=1}^{\kappa C} t_i$, and the expected number of trials is $E[T_\kappa] = \sum_{i=1}^{\kappa C} C/(C-i+1)$ which then gives $E[T_\kappa] = C(1/C + \ldots + C/(C-\kappa C+1)) = C(\mathcal{H}(C) - \mathcal{H}(C-\kappa C))$ where $\mathcal{H}(N) = 1 + 1/2 + \ldots + 1/N$.

It is well-known that $\ln n \leq \mathcal{H}(n) \leq \ln n + 1$ [16] and $\mathcal{H}(n) - \mathcal{H}(\alpha n)$ approaches $\ln n - \ln \alpha n = -\ln \alpha$ as n grows. This gives $E[T_\kappa] \approx -C \ln(1-\kappa)$. Thus for fixed $\kappa < 1$, only $O(C)$ trials, on average, are needed to collect κC coupons. In the case of error-free encoded NW logic decoders this bounds the *expected number* of NWs required. We now bound the number of NWs required with probability $1-\epsilon$.

Lemma IV.1 *Consider the classic coupon collector problem in which one of C coupons is collected independently at random during each of T trials, and each coupon is collected with equal probability. Let S_C denote the number of distinct coupons collected after $T = C$ trials. For $0 > \kappa > 1$*

$$P[S_C < \kappa C] \leq (e^{-\delta}/(1-\delta)^{1-\delta})^{C(1-\kappa+\kappa^2/2)}$$

where $\delta = (1-\kappa/(1-\kappa+\kappa^2/2))$.

Proof: Let x_i be a 0-1 random variable that denotes whether a new coupon is collected during the i^{th} trial. Notice that $p(x_i = 1) \geq (C-i+1)/C$, since by the i^{th} trial at most $i-1$ coupons have already been collected. Furthermore, if fewer than κC coupons have been collected by the i^{th} trial, $p(x_i = 1) > 1-\kappa$.

We wish to bound the probability that $S_C = \sum_{i=1}^{C} x_i < \kappa C$. To do this, we instead consider the sum of C *independent* 0-1 random variables, y_i. Here $p(y_i = 1) = (C-i+1)/C$ for $i \leq \kappa C$ and $p(y_i = 1) = 1-\kappa$ for $i > \kappa C$. Let $S'_C = \sum_{i=1}^{C} y_i$. By the logic of the previous paragraph, $P[S_C < \kappa C] \leq P[S'_C < \kappa C]$.

Since S'_C is the sum of independent random variables, $P[S'_C < \kappa C]$ can be bounded using a Chernov bound, $Pr(S'_C \leq (1-\delta)E[S'_C]) \leq (e^{-\delta}/(1-\delta)^{1-\delta})^{E[S'_C]}$, where $\delta = 1-\kappa C/E[S'_C]$. Here $E[S'_C] = \sum_{i=1}^{C} E[y_i] = \sum_{i=1}^{\kappa C}(C-i+1)/C + \sum_{i=\kappa C+1}^{C} 1-\kappa > C(1-\kappa/2)\kappa + C(1-\kappa)(1-\kappa) = C(1-\kappa+\kappa^2/2)$. This gives $\delta = 1-\kappa/(1-\kappa+\kappa^2/2)$, the desired result. ■

As an example, we can use the above lemma to consider an error-free encoded NW decoder in which $M = N = (5/2)N_A$, in which case $\kappa = 2/5$, $1-\kappa+\kappa^2/2 = 17/25$ and $(1-\kappa/(1-\kappa+\kappa^2/2)) = 7/17$. This gives $P[S_C < 2/5C] \leq (e^{-7/17}/(10/17)^{10/17})^{17C/25}$ which is less than 0.0131 when $C = N_A \geq 64$. As N_A increases further, a larger value of κ can be used. When $\kappa = 1/2$, we have $1-\kappa+\kappa^2/2 = 5/8$ and $(1-\kappa/(1-\kappa+\kappa^2/2)) = 1/5$. This gives $P[S_C < 1/2C] \leq (e^{-1/5}/(4/5)^{4/5})^{5C/8}$ which is less than .01 when $C \geq 343$.

Thus if we consider an error-free encoded NW decoder where $M = N = \beta N_A$, and choose β such that a set of at least N_A MWs exists with probability $1 - 0.0131$, then $MN < (5/2)^2 N_A^2$ when $N_A \geq 64$, and $MN < 2^2 N_A^2$ when $N_A \geq 343$. This outperforms the deterministic construction, for which $MN > w^2 N_A^2$, if $w > (2.5)^2 = 6.25$ in the first case and $w > 4$ in the second case. Lemma IV.1 also yields the following asymptotic result.

Theorem IV.1 *Consider an error-free encoded NW decoder with M MWs and N NWs using $(1, M)$-hot encodings. For any $\epsilon > 0$ and $\beta > 1/(2-\sqrt{2})$ there exists a threshold $N_{\epsilon,\beta}$, such that if $N_A \geq N_{\epsilon,\beta}$ and $M = N = \beta N_A$, then there exists uniquely coupled sets of N_A MWs and N_A NWs with probability at least $1-\epsilon$.*

Proof: In the encoded NW decoder, each NW is controlled by exactly one randomly selected MW. As such, each NW can be thought of as collecting one of $C = M = \beta N_A$ coupons independently at random and with equal probability. We wish to guarantee that at least C/β distinct coupons are collected among the $N = C$ independent trials, given that each trial collects each coupon with probability $1/C$.

Let S_C denote the number of distinct coupons collected after C trials, and let $\kappa = 1/\beta$. In the proof of Lemma IV.1, it is shown that $Pr(S_C \leq (1-\delta)E[S'_C]) \leq (e^{-\delta}/(1-\delta)^{1-\delta})^{E[S'_C]}$, where $E[S'_C] = C(1-\kappa+\kappa^2/2)$ and $\delta = 1-\kappa C/E[S'_C]$. This implies that for fixed $\delta > 0$, $Pr(S_C \leq (1-\delta)E[S'_C]) \to 0$ as C increases.

Requiring that $\delta > 0$ is equivalent to requiring that $1 - \kappa/(1-\kappa+\kappa^2/2) > 0$, or $1 > \kappa/(1-\kappa+\kappa^2/2)$. This becomes $\kappa^2 - 4\kappa + 2 > 0$, which holds when $\kappa < 2-\sqrt{2}$. ■

The bounds on β given in the above Theorem apply to an encoded NW decoder that is error free. If misalignment errors occur with probability p_f, then for any particular value of N_A, the corresponding bound on β scales by a factor of at most $1/(1-p_f)$. To see why, notice that in the proof of Lemma IV.1, $E[x_i]$ and $E[y_i]$ are scaled by a factor of $1/(1-p_f)$, as are $E[S_C]$ and $E[S'_C]$. This same scaling bound on β applies to RCDs, for which $p_f \approx 1-\alpha e^{-1}$ when $p = \alpha/N$. In this case p_f denotes the probability that a particular NW's codeword fails to be $(1, M)$-hot. This overly strict condition on codewords is acceptable since we are upper bounding β.

V. Lower Bounding β

The previous section demonstrated that setting $M = N = \beta N_A$, where $\beta > 1/((1-p_f)(2-\sqrt{2}))$, allows for a logic decoder with N_A outputs and area $O(MN) = O(\beta^2 N_A^2)$. This section gives an information theoretic lower bound on β as N_A grows. Our general approach can potentially be applied to other stochastically assembled structures as well.

To begin, let the **configuration**, $\mathcal{C}$, of a decoder denote the state of its MN MW/NW junctions (i.e. codewords $\boldsymbol{c^1} \dots \boldsymbol{c^N}$). A configuration is **successful** if it contains sets of N_A NWs and N_A MWs, $\mathcal{N}$ and $\mathcal{M}$, that are uniquely coupled. When a NW decoder is stochastically assembled, let $1-\epsilon$ be the probability that the resulting configuration is successful.

The basic approach used to lower bound β is relatively straightforward. Before a NW decoder is assembled, there is a probability distribution associated with $\mathcal{C}$. Depending on the parameters of the assembly process, there is a certain amount of entropy (i.e. uncertainty), denoted $h(\mathcal{C})$, associated with $\mathcal{C}$. For RCDs and encoded NW decoders $h(\mathcal{C})$ is easy to compute. Given β, it is possible to upper bound the entropy of $\mathcal{C}$ given that $\mathcal{C}$ is successful. This bound implies a lower bound on β. Specifically, β must be large enough so any upper bound on the entropy of all successful configurations is at least $(1-\epsilon)h(\mathcal{C})$.

More formally, imagine the repeated assembly of a stochastically assembled NW decoder with M MWs and N NWs. Here $h(\mathcal{C})$ represents the minimum number of bits, on average, required to specify $\mathcal{C}$ after each assembly process, among all possible configurations. In other words, suppose that after each decoder is assembled its configuration, $\mathcal{C}$, is recorded in binary using a predetermined encoding scheme. For any such scheme, the average number of bits required per decoder is at most $h(\mathcal{C})$. Also, the bound is asymptotically achievable [17].

If $\mathcal{C}$ is successful with probability $1-\epsilon$, Shannon's source coding theorem implies that as ϵ shrinks and $MN = \beta N_A^2$ increases, the entropy of $\mathcal{C}$, when restricted to only successful configurations, approaches $(1-\epsilon)h(\mathcal{C})$ [17]. This in turn implies that for arbitrarily large values of N_A, the average number of bits required by an encoding scheme that describes only successful configurations is at least $(1-\epsilon)h(\mathcal{C})$.

In a successful decoder, let $\mathcal{S}$ denote the set of the N_A^2 junctions of the uniquely coupled sets $\mathcal{M}$ and $\mathcal{N}$. Also let $\mathcal{C}-\mathcal{S}$ denote the set of the remaining $MN - N_A^2$ junctions. To obtain a lower bound on β, we observe that the average number of bits required to specify a successful configuration is at most the average number of bits required to specify $\mathcal{S}$, denoted $h(\mathcal{S})$, plus the average number of bits required to specify $\mathcal{C}-\mathcal{S}$ given $\mathcal{S}$, denoted $h(\mathcal{C}-\mathcal{S}|\mathcal{S})$. As explained above, the average number of bits required to specify a successful configuration is at least $(1-\epsilon)h(\mathcal{C})$. Thus

$$(1-\epsilon)h(\mathcal{C}) \leq h(\mathcal{S}) + h(\mathcal{C}-\mathcal{S}|\mathcal{S}) \tag{1}$$

which we now apply to both RCDs and encoded NW decoders. For simplicity we consider the bound (which holds for all $\epsilon > 0$) as $\epsilon \to 0$. This gives:

$$h(\mathcal{C}) \leq h(\mathcal{S}) + h(\mathcal{C}-\mathcal{S}|\mathcal{S}) \tag{2}$$

A. A Lower Bound for RCDs

To lower bound β for RCDs we can assume decoders are error-free. Here $\boldsymbol{c^i_j} = 1$ with probability p and $\boldsymbol{c^i_j} = 0$ with probability $1-p$. The $\boldsymbol{c^i_j}$ are independent random variables, so $h(\mathcal{C}) = MNh(p)$, where $h(p) = -p\log p - (1-p)\log(1-p)$ is the binary entropy function [17] (log is base 2).

$h(\mathcal{S})$ is upper bounded below. $h(\mathcal{C}-\mathcal{S}|\mathcal{S}) \leq (MN - N_A^2)h(p^*)$, where p^* is the probability that a given junction in $\mathcal{C}-\mathcal{S}$ is controlling. From inequality 2, this gives

$$MNh(p) \leq h(\mathcal{S}) + (MN - N_A^2)h(p^*) \tag{3}$$

which implies a bound on β in terms of ϵ and N_A, since $h(\mathcal{S})$ and p^* are both functions of β, N_A and ϵ.

To compute p^*, note that in $\mathcal{C}-\mathcal{S}$ exactly N_A controlling junctions are removed from $\mathcal{C}$. Thus $p^* = (MNp - N_A)/(MN - N_A^2)$. Since $MN = \beta^2 N_A^2$ this gives $p^* = (\beta^2 N_A^2 p - N_A)/(N_A^2(\beta^2-1)) = (\beta^2 p - 1/N_A)/(\beta^2-1)$. Thus for fixed p, p^* approaches $p\beta^2/(\beta^2-1)$ as N_A increases. Also $p \leq p^* \leq p\beta^2/(\beta^2-1)$ when $p \geq 1/N_A$.

The average number of bits required to specify $\mathcal{S}$, $h(\mathcal{S})$, is at most the number of bits required to specify $\mathcal{N}$ and $\mathcal{M}$ plus the number required to give an ordering of one of the sets (this specifies which MW is coupled to each NW). This requires $\log\binom{M}{N_A} + \log\binom{N}{N_A} + \log N_A!$ bits. Inequality 3 becomes

$$MNh(p) \leq \log\binom{M}{N_A} + \log\binom{N}{N_A} + \log N_A! + (MN - N_A^2)h(p^*)$$

Stirling's approximation tells us $\log N_A!$ rapidly approaches $N_A \log N_A - N_A \log e + \frac{1}{2}\log(2\pi N_A)$. This implies that $\log\binom{\beta N_A}{N_A} = \log(\beta N_A)! - \log(\beta N_A - N_A)! - \log N_A! < \beta N_A h(1/\beta)$. From this we get $MNh(p) - MNh(p^*) + N_A^2 h(p^*) \leq 2\beta N_A h(1/\beta) + N_A \log N_A - N_A \log e + \frac{1}{2}\log(2\pi N_A)$. Since $MN = \beta^2 N_A^2$ we have

$$N_A\left(\beta^2 h(p) - (\beta^2-1)h(p^*)\right) \leq 2\beta h(1/\beta) + \gamma(N_A) \tag{4}$$

where $\gamma(N_A) = \log N_A - \log e + \frac{1}{2N_A}\log(2\pi N_A)$. This implies a lower bound β given N_A and p. To obtain an explicit bound, we show the implied bound is weakest when p is small.

Observe that as N_A increases, $p \to 0$ if β remains constant (i.e. it is not possible for $MN = O(N_A^2)$ unless $p \to 0$). To see why, notice that for fixed β the right-hand side of the above inequality increases logarithmically in N_A. The left-hand side, however, increases linearly in N_A unless the coefficient $\beta^2 h(p) - (\beta^2-1)h(p^*)$ goes to zero. This implies that as N_A increases, $\beta^2 h(p) - (\beta^2-1)h(p^*)$ must approach 0 for the inequality to hold. Since $p^* \to p$ as N_A increases, $h(p^*) \to h(p)$ and $\beta^2 h(p) - (\beta^2-1)h(p^*) \to h(p)$. Thus as N_A increases $h(p)$, and hence p, must go to 0.

Having established that p goes to zero as N_A increases, we now consider two cases: $p \geq 1/N_A$ and $p \leq 1/N_A$. The second case is considered below. In the first case $p \leq p^* \leq p\beta^2/(\beta^2-1)$ and as we now show, the expression $C(p) = \beta^2 h(p) - (\beta^2-1)h(p^*)$ is smallest when $p = 1/N_A$.

When $p = 1/N_A$ we have $p^* = p$ and $C(p) = h(1/N_A)$. Since $p^* = (\beta^2 p - 1/N_A)/(\beta^2-1)$, $\frac{dp^*}{dp} = \beta^2/(\beta^2-1)$. Now consider the derivative of $C(p)$. $C'(p) = \beta^2 h'(p) - \frac{dp^*}{dp}(\beta^2 - 1)h'(p^*) = \beta^2 h'(p) - \beta^2 h'(p^*) = \beta^2(h'(p) - h'(p^*))$. Here $h(p) = -p \log p - (1-p)\log(1-p)$ and $h'(p) = \log((1-p)/p)$, so $h'(p) > h'(p^*)$ when $p < p^* < 1/2$. Thus for $p > 1/N_A$, $C'(p) > 0$ and when $p \geq 1/N_A$, $C(p)$ is smallest when $p = 1/N_A$. Inequality 4 now becomes

$$N_A h(1/N_A) \leq 2\beta h(1/\beta) + \log N_A - \log e + (1/2N_A)\log(2\pi N_A)$$

Since $\log(1 - 1/N_A) \leq -1/N_A$, we have $h(1/N_A) \geq 1/N_A \log N_A + 1/N_A - 1/N_A^2$ and thus

$$1 + \log e \leq 2\beta h(1/\beta)$$

which reveals that $\beta > 1.25$ if $p \geq 1/N_A$ and N_A increases.

Finally, we return to the case when $p \leq 1/N_A = \beta/N$. The probability that a NW isn't controlled by any MW is $(1 - \beta/N)^N$, which rapidly approaches $e^{-\beta}$ as N_A, and hence N, increases. Since on average $e^{-\beta}$ NWs are not controlled by any MW, $(1 - e^{-\beta})N \geq N_A$ for a unique coupling to exist with high probability. Since $N = \beta N_A$ we have $(1 - e^{-\beta})\beta \geq 1$, which implies that $\beta > 1.349$. For an RCD with $M = N = \beta N_A$, we have demonstrated that $\beta > 1.25$ as $\epsilon \to 0$.

B. A Lower Bound for Encoded NW Decoders

Now consider error-free encoded NW decoders using $(h,\ M)$-hot codes. Since each codeword is equally likely, $h(\mathcal{C}) = N \log \binom{M}{h}$. As for RCDs, $h(\mathcal{S}) \leq \log \binom{M}{N_A} + \log \binom{N}{N_A} + \log N_A!$. Finally $h(\mathcal{C} - \mathcal{S}) \leq (N - N_A)\log \binom{M}{h} + N_A \log \binom{M}{h-1}$. As $\epsilon \to 0$, inequality 2 yields

$$N_A \log \binom{M}{h} - N_A \log \binom{M}{h-1} \leq h(\mathcal{S}) \qquad (5)$$

From this we have $\log \binom{M}{h} - \log \binom{M}{h-1} \leq 2\beta h(1/\beta) + \log N_A - \log e + \frac{1}{2N_A}\log(2\pi N_A)$. Since $\log \binom{M}{h} - \log \binom{M}{h-1} = \log \binom{M}{h} / \binom{M}{h-1} = \log(M - h + 1)/h$, we have $\log(\beta N_A - h + 1)/h - \log N_A + \log e \leq 2\beta h(1/\beta) + \frac{1}{2N_A}\log(2\pi N_A)$. Finally, since $\log(\beta N_A - h + 1)/h - \log N_A = \log(\beta/h - 1/N_A + 1/hN_A)$, as N_A increases we have

$$\log(\beta/h) + \log e \leq 2\beta h(1/\beta) \qquad (6)$$

If $h = 1$, this becomes $\log e \leq 2\beta h(1/\beta) - \log \beta$, which implies that $\beta > 1.24$. When $h = 2$ or more, $\log(\beta/h)$ changes sign and the bound becomes quite weak. Still, as N_A increases, we would not expect $h \geq 2$ to outperform $h = 1$.

VI. Conclusion

In this paper we have extended the type of modeling and analysis applied previously to stochastically assembled NW memory decoders. This has enabled us to bound the area of stochastically assembled NW logic decoders. The analysis of Section IV shows that the overhead associated with stochastic assembly of logic decoders is at most a small constant factor. This is encouraging, since similar results have been obtained for memory decoders. These results collectively suggest that the overhead associated with stochastic assembly of functional nanoscale devices is generally modest. Additionally, Section V introduces a novel information theoretic approach to lower bounding this overhead. This same approach is potentially applicable to other stochastically assembled structures.

Our analysis of NW logic decoders is based on the condition that a decoder must contain uniquely coupled sets of N_A NWs and N_A MWs, with high probability. Interestingly, this is the same condition required for stochastically assembled inversion and buffering layers within NW crossbar logic. Here two sets of NWs are placed at right angles, and randomly placed FETs cause each input NW in one dimension to gate one or more randomly chosen output NWs in the second dimension (see Figure 2). As described by DeHon in [12], this type of stochastically assembled structure provides signal restoration, and (optionally) logical negation. If we wish to be able to restore at least N_A inputs, we once again wish to identify N_A input wires that are uniquely coupled to N_A output wires. As such, the analysis of the previous sections applies.

References

[1] N. A. Melosh, A. Boukai, F. Diana, B. Gerardot, A. Badolato, P. M. Petroff, and J. R. Heath, "Ultrahigh-density nanowire lattices and circuits," *Science*, vol. 300, pp. 112–115, Apr. 4, 2003.

[2] R. Beckman, E. Johnston-Halperin, Y. Luo, J. E. Green, and J. R. Heath, "Bridging dimensions: Demultiplexing ultrahigh-density nanowire circuits," *Science*, vol. 310, pp. 465–468, 2005.

[3] Z. Zhong, D. Wang, Y. Cui, M. W. Bockrath, and C. M. Lieber, "Nanowire crossbar arrays as address decoders for integrated nanosystems," *Science*, vol. 302, pp. 1377–1379, 2003.

[4] D. Whang, S. Jin, Y. Wu, and C. M. Lieber, "Large-scale hierarchical organization of nanowire arrays for integrated nanosystems," *Nano Letters*, vol. 3, no. 9, pp. 1255–1259, 2003.

[5] J. E. Savage, E. Rachlin, A. DeHon, C. M. Lieber, and Y. Wu, "Radial addressing of nanowires," *J. Emerg. Technol. Comput. Syst.*, vol. 2, no. 2, pp. 129–154, 2006.

[6] E. Rachlin and J. E. Savage, "Nanowire addressing with randomized-contact decoders," *Theor. Comp. Sci.*, vol. 408, no. 2-3, pp. 241–261, 2008.

[7] X. Ma, D. B. Strukov, J. H. Lee, and K. K. Likharev, "Afterlife for silicon: Cmol circuit architectures," in *Procs. IEEE-NANO*, 2005.

[8] N. Di Spigna, D. Nackashi, C. Amsinck, S. Sonkusale, and P. Franzon, "Deterministic nanowire fanout and interconnect without any critical translational alignment," *IEEE Trans. Nanotechnology*, vol. 5, pp. 356–361, July 2006.

[9] K. Gopalakrishnan, R. S. Shenoy, C. Rettner, R. King, Y. Zhang, B. Kurdi, L. D. Bozano, J. J. Welser, M. B. Rothwell, M. Jurich, M. I. Sanchez, M. Hernandez, P. M. Rice, W. P. Risk, and H. K. Wickramasinghe, "The micro to nano addressing block," in *Procs. IEEE Int. Electron Devices Mtng.*, Dec. 2005.

[10] B. Gojman, E. Rachlin, and J. E. Savage, "Evaluation of design strategies for stochastically assembled nanoarray memories," *J. Emerg. Technol. Comput. Syst.*, vol. 1, no. 2, pp. 73–108, 2005.

[11] E. Rachlin and J. E. Savage, "Analysis of mask-based nanowire decoders," *IEEE Trans. Comput.*, vol. 57, no. 2, pp. 175–187, 2008.

[12] A. DeHon, "Nanowire-based programmable architectures," *J. Emerg. Technol. Comput. Syst.*, vol. 1, no. 2, pp. 109–162, 2005.

[13] T. Hogg, Y. Chen, and P. J. Kuekes, "Assembling nanoscale circuits with randomized connections," *IEEE Trans. Nanotechnology*, vol. 5, no. 2, pp. 110–122, 2006.

[14] A. Dehon, "Deterministic addressing of nanoscale devices assembled at sublithographic pitches," *IEEE Trans. Nanotechnology*, vol. 4, no. 6, pp. 681–687, 2005.

[15] E. Rachlin, *Reliable Computing at the Nanoscale*. PhD dissertation, Brown University, Department of Computer Science, 2010.

[16] M. Mitzenmacher and E. Upfal, *Probability and Computing: Randomized Algorithms and Probabilistic Analysis*. Cambridge University Press, 2005.

[17] D. J. C. MacKay, *Information Theory, Inference, and Learning Algorithms*. Cambridge University Press, 2003.