We propose a method to determine single hyperspace vectors (product strings of noise-bits) by classical means with the same effectiveness as the results using time shifted noise-based logic. A system of binary linear equations based on the amplitudes of the hyperspace vector and the reference noise-bits is set up and solved after enough independent information is collected. The resulting error probability (the chance of getting no answer) has approximately an exponential decay with the time of measurement.

1. Introduction

Noise-based logic (NBL) [1-9], on the contrary of its name, is a deterministic logic scheme. The logic information is carried by independent stochastic processes, their products and their superpositions [1]. In non-squeezed NBL, $2^N$ independent reference noises are needed to form $N$ noise-bits: one for each $L$ and another one for each $H$ value and $N$-bit long products (product-strings) represent hyperspace vectors.

The identification of a single product-string (hyperspace vector) is necessary after a quantum-computing-type calculation with a simple output (such as the binary representation of a prime number) and it is one of the challenging questions. Recently, "time shifted noise-based logic" has been introduced [9], which is constructed by shifting each reference signal with a small time delay. This modification implies an exponential speedup of single hyperspace vector identification compared to the former case and it requires the same, $O(N)$ complexity as in quantum computing.

The goal of the present Letter is to show that the identification of such a single hyperspace vector can be done with similar complexity without the need of time shifted noise-based logic. Thus this task can be carried out within the original noise-based logic framework, which uses the same clock timing for each reference component.
2. Problem setting

Let $N$ denote the number of bits and for $k = 1, 2, \ldots, N$ let

$$A_k, B_k : \{1, 2, \ldots\} \to \{-1, 1\}$$

be random telegraph waves (RTW). As for the physical realization, each function $A_k$ or $B_k$ represents a sequence of electric impulses with intensity $A_k(t)$ (or $B_k(t)$) in every time interval $(t - 1, t)$ between consecutive clock signals for $t = 1, 2, \ldots$.

We are given a binary string $x_0, x_1, \ldots, x_N$ consisting of terms 0 or 1 as usually which will be transformed into the product wave

$$W := X_1X_2\cdots X_N \text{ where } X_k = A_k \text{ if } x_k = 1 \text{ and } X_k = B_k \text{ if } x_k = 0. \quad (2)$$

**Problem 0.** (Measurement problem). Knowing only the product wave $W$, determine the 0-1 string $x_1, \ldots, x_N$ satisfying (2).

Our purpose will be to develop an algorithm based on an infinite system of binary linear algebraic equations describing the relationships encoded in (2). Given the waves $W, A_1, B_1, A_2, B_2, \ldots, A_N, B_N$, for $T = 1, 2, \ldots$, our algorithm reads the impulses $W(T), A_1(T), B_1(T), \ldots, A_N(T), B_N(T)$ and it determines if there is only a unique string $x_1, \ldots, x_N$ suitable for solving Problem 0 on the basis of the gathered data as far. Having found the first time

$$S = S(W, A_1, B_1, \ldots, A_N, B_N) \quad (3)$$

with this uniqueness property, the algorithm provides the precise guess for the sequence $x_0, x_1, \ldots, x_N$ and then stops. We shall call $S$ the time requirement for complete measurement. In Definition 3 below we formulate a precise algebraic construction for $S$. Notice that the possibility $S = \infty$ cannot be excluded as e.g. if $W(t) = A_k(t) = B_k(t) \equiv 1$. However, such algebraically extreme cases are practically improbable due to the random construction of the waves $A_k, B_k$. Actually, throughout the whole paper we regard the terms $A_k(t), B_k(t)$ ($k = 1, \ldots, N; t \in \mathbb{Z}_+$) to be completely independent random variables such that

$$\Prb(A_k(t) = (-1)^s) = \Prb(B_k(t) = (-1)^s) = \frac{1}{2} \quad (s = 0, 1). \quad (4)$$

**Definition 1.** Introduce the below transformed variables resp. product wave:

$$C_k := A_k B_k \quad (k = 1, \ldots, N), \quad U := WB_1B_2\cdots B_N. \quad (5)$$

**Observation 1.** Due to the identities $A_k^2 = B_k^2 = 1$, we have $X_k = C_k^{s_k}B_k$, and hence $X_1X_2\cdots X_N = W \iff (X_1B_1)(X_2B_2)\cdots (X_NB_N) = U$ that is

$$C_1^sC_2^s\cdots C_N^s = U. \quad (6)$$

**Definition 2.** Introduce the matrix $C := \begin{bmatrix} c_{tk} \end{bmatrix}_{t=1}^{\infty} \quad (k=1) = 1$ respectively the column vector $u := \begin{bmatrix} u_t \end{bmatrix}_{t=1}^{\infty}$ as follows: we let

$$c_{tk} := \begin{cases} 1 & \text{if } C_k(t) = -1, \\ 0 & \text{if } C_k(t) = 1 \end{cases}, \quad (7)$$

$$u_t := \begin{cases} 1 & \text{if } U(t) = -1, \\ 0 & \text{if } U(t) = 1 \end{cases}. \quad (8)$$
Observation 2. For any time $t$ we have 

$$(-1)^{u(t)} = U(t) = C_1(t)x_1 \cdots C_N(t)x_N = (-1)^{c_{11}x_1 + \cdots + c_{N1}x_N}.$$ 

Hence the relationship $U = C_1^{x_1} \cdots C_N^{x_N}$ is equivalent to the following (infinite) system of linear equations over the field $\mathbb{Z}_2 := \{0, 1\}$ of two elements 

$$c_{11}x_1 + c_{12}x_2 + \cdots + c_{1N}x_N = u_1$$
$$c_{21}x_1 + c_{22}x_2 + \cdots + c_{2N}x_N = u_2$$
$$\vdots \quad \vdots \quad \vdots \quad \vdots$$
$$c_{t1}x_1 + c_{t2}x_2 + \cdots + c_{tN}x_N = u_t.$$ 

Remark 1. The answer to Problem 0 is nothing else than to find the solution $x_1, \ldots, x_N \in \mathbb{Z}_2$ of the above infinite system of linear equations. Notice that this system has a solution by assumption. On the other hand, for $T \to \infty$ the columns of the submatrix $C^{(T)} := [c_{tk}]_{t=1}^{\infty} \in \mathbb{Z}_2$, $1 \leq k \leq N$ become linearly independent (over $\mathbb{Z}_2$) with probability 1 since the variables $C_k(t)$ $(1 \leq t \leq T, 1 \leq k \leq N)$ are completely independent in stochastic sense.

Definition 3. Let $S$ denote the minimal index for which the columns of $C^{(S)}$ are linearly independent (over $\mathbb{Z}_2$). Notice that $S$ is also a random variable with $\lim_{s \to \infty} \text{Prb}(S > s) = 0$. Henceforth let us write 

$$\pi_s := \text{Prb}(S = s), \quad \pi_t := \text{Prb}(S \geq t).$$ 

Notice that $\pi_t$ and $\pi_t$ are the probabilities of for our measurement algorithm to stop exactly at time $t$ and to fail before time $t$, respectively. Actually we have $\pi_t = \sum_{s=t}^{\infty} \pi_s$.

Problem 1. Find sharp asymptotic estimates for $\pi_t$.

Problem 2. Considering large values $N$ of bits, is it possible to find a function $N \mapsto t_N$ such that $t_N << 2^N$ and $\pi_{t_N}$ tends to zero with exponential rate as $N \to \infty$.

Remark 2. $2^N$ is the lowest time limit $T$ for the complete orthogonality of the product functions $C_1^{x_1} \cdots C_N^{x_N}$ when restricted to the interval $\{1, \ldots, T\}$.

3. Results

To answer Problems 1-2, we are going to investigate the Gaussian elimination algorithm applied to the matrix $C$ establishing row indices $t_1 < t_2 < \cdots < t_N$ such that row $t_1$ is the first non vanishing row in $C$ and, for each $k = 2, \ldots, N$, row $t_k$ is the least index with row $t_k$ being linearly independent of rows $t_1, \ldots, t_{k-1}$ in $C$.
To do so, we can proceed to find entries with indices $(t_1, k_1), (t_2, k_2), \ldots, (t_N, k_N)$ in the following manner.

**Algorithm.**

Let $G_0 := C$ the matrix introduced in Definition 2, $K_0 := \{1, \ldots, N\}$.

- **Step 1)** Let $t_1$ be the row index of the first nonzero row in $G_0$ and let $k_1$ be the column index of the first non-zero entry in this row. Then we set $K_1 := K_0 \setminus \{k_1\}$ and we let $G_1$ to be the matrix obtained with elimination killing the subcolumn below the entry $(t_1, k_1)$.
- **Step $d$)** Let $t_d$ be the first row index $> t_d-1$ such that row $t_d$ in $G_{d-1}$ does not vanish but all the rows of of $G_{d-1}$ vanish between row $t_d-1$ and $t_d$. Then let $k_d$ be the minimal index in $K_{d-1}$ such that the entry of $G_{d-1}$ with index $(t_d, k_d)$ does not vanish. We define $K_d := K_{d-1} \setminus \{k_d\}$ and the matrix $G_d$ is obtained with elimination from $G_{d-1}$ by killing the subcolumn below the entry $(t_d, k_d)$.

**Theorem 1.** Let $0 = t_0 < t_1 < t_2 < \cdots < t_N$ be arbitrarily given integers and let $n_d := t_d - t_{d-1} - 1$ $(d = 1, \ldots, N)$. Then we have

$$\Prb{\text{the Algorithm produces the row indices } t_1, \ldots, t_N} = \prod_{d=0}^{N-1} \left[ 2^n_d(N-d)(1-2^{d-N}) \right].$$

**Proof.**

$$\Prb{\text{the Algorithm produces the row indices } t_1, \ldots, t_N} =$$

$$= \left[ \Prb{\text{the } Nn_0 \text{ entries of the first } n_0 \text{ rows vanish}} \right] \times$$

$$\times \sum_{r=0}^{N-1} \Prb{r \text{ zeros stand before a term } 1 \text{ in row } t_1} \times$$

$$\times \left[ \Prb{\text{the } (N-1)n_1 \text{ entries of the first } n_1 \text{ rows after the } t_1 \text{th vanish}} \right] \times$$

$$\times \sum_{r=0}^{N-2} \Prb{r \text{ zeros before a } 1 \text{ in the part with column indices in } K_1 \text{ of row } t_2} \times$$

$$\times \left[ \Prb{\text{the } (N-1)n_1 \text{ entries of the first } n_1 \text{ rows after the } t_1 \text{th vanish}} \right] \times$$

$$\times \sum_{r=0}^{N-2} \Prb{r \text{ zeros before a } 1 \text{ in the part with column indices in } K_1 \text{ of row } t_2}. $$
Since the entries of $C$ assume the values 0-1 with probability $1/2$ independently, hence
\[
\Prb\left(\text{the Algorithm produces the row indices } t_1, \ldots, t_N \right) = \\
= \prod_{d=0}^{N-1} \left[ 2^{-n_d(N-d)} \left(2^{-1} + 2^{-2} + \cdots + 2^{d-N} \right) \right] = \\
= \prod_{d=0}^{N-1} \left[ 2^{-n_d(N-d)} \left(1 - 2^{d-N} \right) \right].
\]

**Corollary.** For any $s \geq N$ we have $\pi_s \leq \left( \frac{s}{N} \right) 2^{N-s}$.

**Proof.** We know that $\pi_s = \sum_{1 \leq t_1, \ldots, t_N = s} \Prb\left(\text{the Algorithm produces the row indices } t_1, \ldots, t_N \right)$. Therefore
\[
\pi_s = \sum_{n_0 + \cdots + n_{N-1} = s-N} \prod_{d=0}^{N-1} \left[ 2^{-n_d(N-d)} \left(1 - 2^{d-N} \right) \right] \leq \\
\leq \sum_{n_0 + \cdots + n_{N-1} = s-N} \exp_2 \left( - \sum_{d=0}^{N-1} n_d(N-d) \right) \leq \\
\leq \sum_{n_0 + \cdots + n_{N-1} = s-N} \exp_2 \left( - \sum_{d=0}^{N-1} n_d \right) = \left( \frac{s}{N} \right) 2^{N-s}
\]
due to the combinatorial fact that $\#\{(n_0, \ldots, n_{N-1}) : n_0 + \cdots + n_{N-1} = s-N\}$ coincides with the number of repeated combination of dividing $s-N$ elements into $N$ parts.

**Theorem 2.** For $t \geq N$ we have $\pi_t \leq 2^{N+1-t} \sum_{k=0}^{N} \left( \begin{array}{c} t \\ k \end{array} \right)$.

**Proof.** Observe that, in general, $\left( \begin{array}{c} s \\ N \end{array} \right) x^{N-s} = \frac{1}{N!} \frac{d^N}{dx^N} x^s$. Thus, by the Corollary,
\[
\pi_t = \sum_{s=1}^{\infty} \pi_s \leq \sum_{s=1}^{\infty} \left( \frac{s}{N} \right)^{N-s} \left( \frac{1}{2} \right)^{N-s} = \frac{1}{N!} \frac{d^N}{dx^N} x^t(1-x)^{-1} \bigg|_{x=1/2}, \quad (11)
\]
According to Leibniz' rule,
\[
\frac{d^N}{dx^N} x^t(1-x)^{-1} = \sum_{k=0}^{N} \left( \begin{array}{c} N \\ k \end{array} \right) \frac{d^k}{dx^k} x^t \frac{d^{N-k}}{dx^{N-k}}(1-x)^{-1} = \\
= \sum_{k=0}^{N} \left( \begin{array}{c} N \\ k \end{array} \right) t(t-1) \cdots (t-k+1)x^{t-k}(1-x)^{N-k}(N-k)!(1-x)^{-(N-k+1)}.
\]
Taking into account that $x^{t-k}(1-x)^{-(N-k+1)}|_{x=1/2} = 2^{k-t+N-k+1} = 2^{N+1-t}$, from (11) we get
\[
\pi_t \leq \frac{1}{N!} \sum_{k=0}^{N} \binom{N}{k} \frac{t!}{(t-k)!} (N-k)! 2^{N+1-t} = \sum_{k=0}^{N} \binom{t}{k} 2^{N+1-t}.
\]

**Corollary.** For $t \geq N$ we have $\pi_t \leq 2(2^N - t)N$.

**Proof.** Putting the inequalities $\binom{t}{k} = \frac{t(t-1)\cdots(t-k+1)}{k!} \leq t^k/k! \leq 1$, we get $\pi_t \leq 2t^N 2^{N-t} \sum_k k^{-1}$. We are now ready to answer Problem 2. Notice that as far we used an arbitrarily fixed number $N$ of bits (and we suppressed the parameter $N$ in the notation in the notation $\pi_t = \pi_t(N)$). We are now interested in finding reasonably small function $N \mapsto t(N)$ such that $\pi_t(N)$ decreases exponentially to zero as $N \to \infty$, or which is the same, we have $\limsup_{N \to \infty} N^{-1} \log_2 \pi_t(N) < 0$ or even $= -\infty$.

**Proposition 1.** Given any small constant $\varepsilon > 0$, with the choice $t(N) := N(\log_2 N)^{1+\varepsilon}$ we have $\lim_{N \to \infty} N^{-1} \log_2 \pi_t(N) = -\infty$.

**Proof.** In this case
\[
\frac{1}{N} \log_2 \pi_t(N) \leq \frac{1}{N} \log_2 \left(2e\left[N(\log_2 N)^{1+\varepsilon}\right]N\cdot2^{N(\log_2 N)^{1+\varepsilon}-N}\right) =
\]
\[
= \frac{1+\log_2 e}{N} + (1+\varepsilon) \log_2 \log_2 N + \log_2 N = \left(\log_2 N\right)^{1+\varepsilon} \to -\infty.
\]

**Remark 3.** In conclusion, whatever small $\varepsilon > 0$ we choose, the probability that the measurement algorithm fails in less than $t_\varepsilon = N(\log_2 N)^{1+\varepsilon}$ time steps is scaling as $2^{-N}$ in the limit of large $N$.

**Acknowledgements**

I would like to express my sincere gratitude to Prof. L. Kish for his valuable suggestions and the stimulating discussions on the topics. The author was supported partially by the project TAMOP-4.2.1/B-09/-1/KONV-2010-0005.

**References**


Fast measurement of hypervectors in noise-based logic


