

Remodelling correlation: A fault resilient technique of correlation sensitive stochastic designs

Shyamali Mitra ^{*}, Sayantan Banerjee, Mrinal Kanti Naskar

Jadavpur University, Kolkata, India

ARTICLE INFO

Keywords:

Stochastic logic circuits
Soft errors
Reliability
Remodelling correlation
Priority-based approach

ABSTRACT

Major sources of error in stochastic circuits are correlation errors, soft errors, and random fluctuation errors, which impact the circuit's accuracy and reliability. The soft error has the effect of modifying the correlation status, which in turn modifies the probability of the output. This has serious implications for security and medical systems that require highly precise systems. To address this issue, we employ a fault-tolerant technique of correlation-sensitive stochastic logic circuits. To ensure reliable operation, we have developed a *Remodelling Correlation (ReCo)* framework for correlation-sensitive stochastic logic elements (SLEs). Using two intriguing case studies, we present two variants of *ReCo* model for combinational circuits with contradictory requirements. To achieve faster convergence to the desired Mean-Squared Error (MSE) value with less hardware area, the proposed method prioritizes the selection of logic elements and the placement of correction blocks. It is shown that the overall reliability of the circuit is unaffected by this method. To demonstrate the usefulness of the proposed framework, the contrast stretch operation on images of CEED2016, a standard contrast enhancement dataset, is investigated in a noisy setting. The average Multiscale structural similarity index (MS-SSIM) of the output images using the proposed method is observed as 0.91 ± 0.02 , which is significantly higher than the original images with error, i.e., 0.75 ± 0.12 .

1. Introduction

Computation on binary numbers using *Stochastic computing* [1] is gaining popularity nowadays because it offers several advantages [2] compared to conventional weighted-binary computation. It is a low power and low cost alternative to complex arithmetic functions. With a remarkable reduction in size, circuit complexity and power consumption, stochastic architecture has proved to be noise immune compared to the conventional implementation of binarization algorithms [3] and various other image processing tasks also [4]. Different types of errors, such as soft errors, correlation induced errors, and random fluctuation errors are identified that affect the accuracy and reliability of stochastic circuits [5]. Thus, to generate the desired function using unreliable components in the presence of errors has become a challenging task. Transient or soft errors are caused due to exposure to external radiation and are greatly increased by manufacturing defects in a chip. These introduce false logic at the output of the circuit [6]. If multiple faults occur at the nodes of a gate, the output may be erroneous. Soft errors are the cause of bit-flips in a stochastic bitstream and may lead to an undesirable correlation between two numbers.

There are several analytical approaches to assess the reliability of probabilistic circuits with unpredictable behaviours; Probabilistic

Gate Models (PGM), Probabilistic Transfer Matrices (PTM), Stochastic Computational Model (SCM), Monte Carlo Simulation [7] etc. Nevertheless to mention that majority of works are aligned towards reliability assessment and analysis [8,9]. In [10], a multiple fault modelling methodology was proposed for efficient estimation of the circuit's reliability. Gates that have the highest impact on circuit's reliability were determined to design trade-off at early stages of circuit design. There are several approaches proposed in literature for modelling and analysis of transient faults in logic circuits [11], modelling and reduction [12], soft error analysis tool [13] for combinational logic circuits. In [14] a Bernoulli distribution based-model for reliability calculation was developed to decompose the evaluation objective of circuits, with single and double fault simulations. The proposed method is scalable with circuit size and is independent of the soft error rate. But the above works deal with the modelling and analysis of soft errors in weighted-binary logic circuits. Recently, Ting et al. [15] investigated the role of constant inputs in SC, and propose an algorithm to eliminate them by introducing sequential circuits. But they do not deal with handling the transient error scenarios in stochastic circuits. To our knowledge, no correction mechanism exists in the literature to generate reliable output under transient error scenarios for stochastic logic circuits.

* Corresponding author.

E-mail address: shyamalimitra.iee@jadavpuruniversity.in (S. Mitra).

The proposed work focuses on the erroneous behaviour of stochastic circuits, which is primarily attributable to transient errors under noisy conditions, and its methodical correction using a correlation-based framework. Correlation is one of the intriguing topics in SC and has become a powerful tool to analyze the difference in behaviour of the same logic circuit at different correlation status. This investigation is focused on determining the behaviour of circuits subjected to soft errors. To reduce the impact of transient faults and ensure the circuit's reliability, bitstreams are injected with desired correlations. Experiments are conducted on SLEs whose functional behaviour vary with changes in correlation. The objective is to develop a technology-independent framework for observing error-free output in complex circuits with minimal hardware. A study is also conducted to demonstrate that the reliability of a circuit is unaffected by subtle changes in correlation status. The contributions are highlighted as follows:

- We develop a correlation-based framework for correlation sensitive SLEs under transient error scenarios to model the error-free output.
- A priority-based approach in the selection of SLEs is explored to reduce the hardware complexity and improve the accuracy in computation for complex circuits.
- Two practical multi-input multi-level circuits with two distinct conditions are considered and treated utilizing two different algorithms.
- Evaluation of the proposed work on contrast stretch operations on images under high transient error rates.

In essence, this work not only contradicts the popular perception with regard to correlation, but firmly establishes that injecting a controlled degree of correlation can improve the error-resilient behaviour of the circuit. The rest of the work is organized as follows: In Section 2, we introduced *PTM* as a tool to assess reliability for correlation sensitive designs. Section 3 discusses two major sources of error in stochastic circuits and introduces the proposed methodology in the noisy environment for correlation-sensitive logic elements. With several initial correlation assumptions, we have shown that a correct operating point of the circuit can be established with a suitably injected correlation that could generate an accurate result under the stated conditions. In Section 4, we extended the idea to simulate complex SLCs to show the effectiveness of the proposed model. Two approaches based on a priority-search model are demonstrated following two distinct conditions. The applicability of the proposed methodology in the context of an image processing task is discussed in Section 5. In Section 6, highlights of the experimental results are jotted down with the pros and cons of the proposed algorithms.

2. Probability transfer matrices

The *PTM*, which is used in analysing probabilistic logic circuits [16] has proved to be a convenient tool in error [17] and reliability analysis [18] of small stochastic circuits. The *PTM* is derived from Ideal Transfer Matrix (*ITM*) of a logic circuit. Each entry in *ITM* $J(i, j)$ of a logic circuit with $i = i_0, i_1, \dots, i_{n-1}$ and $j = j_0, j_1, \dots, j_{m-1}$ depicts the logic behaviour excited by a set of inputs. A 1 is observed in the matrix where a particular input combination generates a 1 in the truth table. It can be observed as a conditional probability matrix so that, $J(i, j) = p(\text{output} = j | \text{input} = i)$, where p represents the conditional probability of a particular output being true given a certain input combination. In Ideal Transfer Matrix (*ITM*), when the gate is assumed to be error free, elements are either 0 or 1, representing exact binary values instead of probabilities.

In *PTM*(M) each entry contains a real value in the interval $[0, 1]$ associated with output error probability information. For large circuits, computation with *PTM* is tedious. But for a circuit with k inputs and

l outputs, a circuit *PTM* is of size $2^k \times 2^l$. *ITM* and *PTM* for a two input AND gate are represented as matrices J and M .

$$J = \begin{bmatrix} y = 0 & y = 1 \\ 1 & 0 \\ 1 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix}$$

$$M = \begin{bmatrix} y = 0 & y = 1 \\ 1 - p_e & p_e \\ 1 - p_e & p_e \\ 1 - p_e & p_e \\ p_e & 1 - p_e \end{bmatrix}$$

The rows in J and M correspond to input combinations 00, 01, 10, 11. Columns correspond to outputs $y = 0, y = 1$. *PTM* for a large circuit can be calculated from the *PTM* of the individual gates and the way they are interconnected, using two basic operations [16]:

- The overall *PTM* of a circuit consisting of p gates with *PTMs* M_1, M_2, \dots, M_p connected in series, is obtained by multiplication of individual *PTMs*; $M_{\text{series}} = M_1 \cdot M_2 \dots M_p$.
- The resultant *PTM* of p gates with *PTMs* M_1, M_2, \dots, M_p connected in parallel is obtained by the Kronecker product or tensor multiplication of individual *PTM*; $M_{\text{parallel}} = M_1 \otimes M_2 \otimes M_3 \dots \otimes M_p$. Kronecker product or tensor multiplication multiplies each element of a matrix A with size $(m \times n)$ with another matrix B of size $(p \times q)$ to give an output matrix C of size $(m \times n) \times (p \times q)$.

Input stochastic signals fed to a combinational circuit can also be represented via *PTM*. We define an input vector of size 1×2^k , where k is the total number of input signals which when multiplied by the overall circuit *PTM*, M_{ckt} gives the output *PTM*. Let X be a random variable with a Bernoulli distribution with a probability of occurrence of 1 given as p_x . X is expressed as a two-element row vector $M = [1 - p_x \quad p_x]$. For a combinational circuit with uncorrelated input bitstreams X and Y having probabilities p_x and p_y and output signal Z having a probability p_z , we can represent the input vector I_{in} using *PTM* as tensored product of two parallel identity signal matrices as,

$$\begin{aligned} I_{in} &= [(1 - p_x) \quad p_x] \otimes [(1 - p_y) \quad p_y] \\ &= [(1 - p_x)(1 - p_y) \quad (1 - p_x)p_y \quad p_xp_y \quad p_xp_y] \text{ which is a 4-element vector.} \end{aligned}$$

The output distribution Z is given by the matrix product of I_{in} and the *PTM* of the circuit M_{ckt} using Eq. (1).

$$Z = I_{in} \cdot M_{\text{ckt}} = [(1 - p_z) \quad p_z] \quad (1)$$

I_{in} can also be written in terms of bit overlaps in the bitstreams

$$I_{in} = [i_0 \quad i_1 \quad i_2 \quad i_3] = [n_{00} \quad n_{01} \quad n_{10} \quad n_{11}]$$

where, i_0, i_1, i_2, i_3 represents the input probabilities p_x and p_y being 00, 01, 10 and 11 respectively. Thus, it can represent correlation between input signals as well. When $\text{SC}(X, Y) = 1$, maximum overlap of the bitstreams is guaranteed [17]. When $p_x > p_y$, the overlap 01 never occurs, so $n_{01} = 0$. The probabilities of the other three overlaps 00, 10 and 11 are respectively given as $(1 - p_x)$, $(p_y - p_x)$ and p_y . We can therefore express the input vector for two maximally correlated inputs as

$$I_{+1} = [(1 - p_x) \quad 0 \quad (p_x - p_y) \quad p_y], \quad p_x > p_y \quad (2)$$

$$= [(1 - p_y) \quad (p_y - p_x) \quad 0 \quad p_x], \quad p_x > p_y \quad (3)$$

Similarly, for two negatively correlated inputs the 11 overlap never occurs if $p_x + p_y \leq 1$. Similarly, 00 overlap will never appear if $p_x + p_y \geq 1$.

$$I_{-1} = [1 - (p_x + p_y) \quad p_y \quad p_x \quad 0], \quad p_x + p_y \leq 1 \quad (4)$$

$$= [0 \quad (1 - p_x) \quad (1 - p_y) \quad (p_x + p_y) - 1], \quad p_x + p_y \geq 1 \quad (5)$$

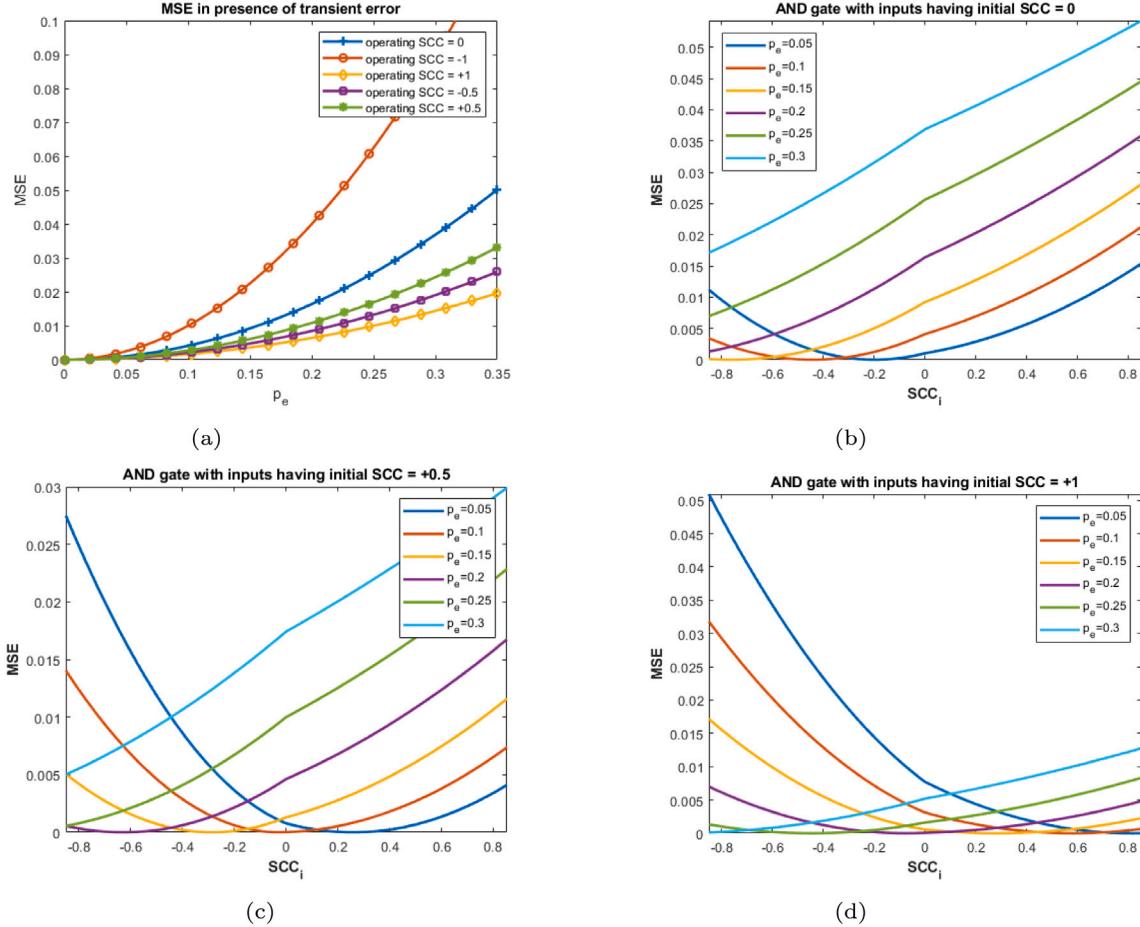


Fig. 1. (a) MSE of AND gate with varying transient errors; Min. *MSE* with *ReCo* for different initial *SCCs* (b) *SCC* = 0 (c) *SCC* = 0.5 (d) *SCC* = 1.

For two uncorrelated numbers the input vector is written using Eq. (6)

$$I_0 = [(1 - p_x)(1 - p_y) \quad (1 - p_x)p_y \quad p_x(1 - p_y) \quad p_xp_y] \quad (6)$$

With the help of Eqs. (2)–(6), we can write the input vector matrix I_{SCC} for any intermediate *SCC* value. These calculations are useful in deriving the stochastic functions given the correlation status between given inputs. *PTMs* are used to study sequential circuits in the same way they are used to study and analyse combinational circuits [18].

2.1. Reliability measure for circuits with varying correlation

We are often concerned with the reliability of circuits under noisy conditions. The reliability of a circuit is defined as its ability to produce a correct output on a regular basis. For stochastic circuits, it can be evaluated using the circuit's *ITM*(*J*) and *PTM*(*M*). It can be shown that the reliability of the circuit does not change with changes in correlation, but rather depends on the probabilistic error in the circuit.

Definition 1. The reliability of a circuit R_{ckt} is invariant to change in correlation between inputs and depends on the probabilistic error rate ' p_e '.

Circuit reliability [18] is a measure of the similarity between its *ITM* and *PTM* and is written as:

$$R_{ckt} = \sum_{J(i,j)=1} p(j|i).p(i) \quad (7)$$

where, $p(j|i)$ is the $(i,j)^{th}$ entry of *PTM*. Using Eq. (7), reliability of the circuit at *SCC* = 0 is obtained as $(1 - p_e)$. For AND gate, R_{ckt} for different ranges of *SCC* can be obtained using Eq. (8)

$$R_{ckt} = \begin{cases} (p_e - 1)(SCC(1 - p_x - p_y + p_x \cdot p_y) + p_x p_y - 1) + \\ (SCC(p_x + p_y - 1) - p_x \cdot p_y(SCC + 1))(p_e - 1) \\ = 1 - p_e, \quad \forall \text{ } SCC < 0 \\ (p_e - 1)(SCC \cdot p_x + p_x \cdot p_y - SCC \cdot p_x \cdot p_y - 1) - \\ (SCC \cdot p_x - p_x \cdot p_y \cdot (SCC - 1))(p_e - 1) \\ = 1 - p_e, \quad \forall \text{ } SCC > 0 \end{cases} \quad (8)$$

Eq. (8) demonstrates that the circuit's reliability is independent of the correlation between the input numbers. Thus, error minimization by varying correlation has no effect on the circuit's reliability. This property aids subsequent analysis of *SLE* in producing the corrected output in the presence of transient error.

3. Major error sources

Circuits are becoming more vulnerable to transient faults as feature sizes, operating voltages, and design margins continue to shrink. Cosmic rays, capacitive coupling, electromagnetic interference, and power transients are some of the leading physical phenomena. Transient faults induced by radiation have gained a lot of attention in recent years because they are seen as a potential roadblock to further technological advancement. In stochastic computing another major challenge is the correlation induced errors that are often associated with reordering of bits caused due to bit flips, which may or may not appear as a change

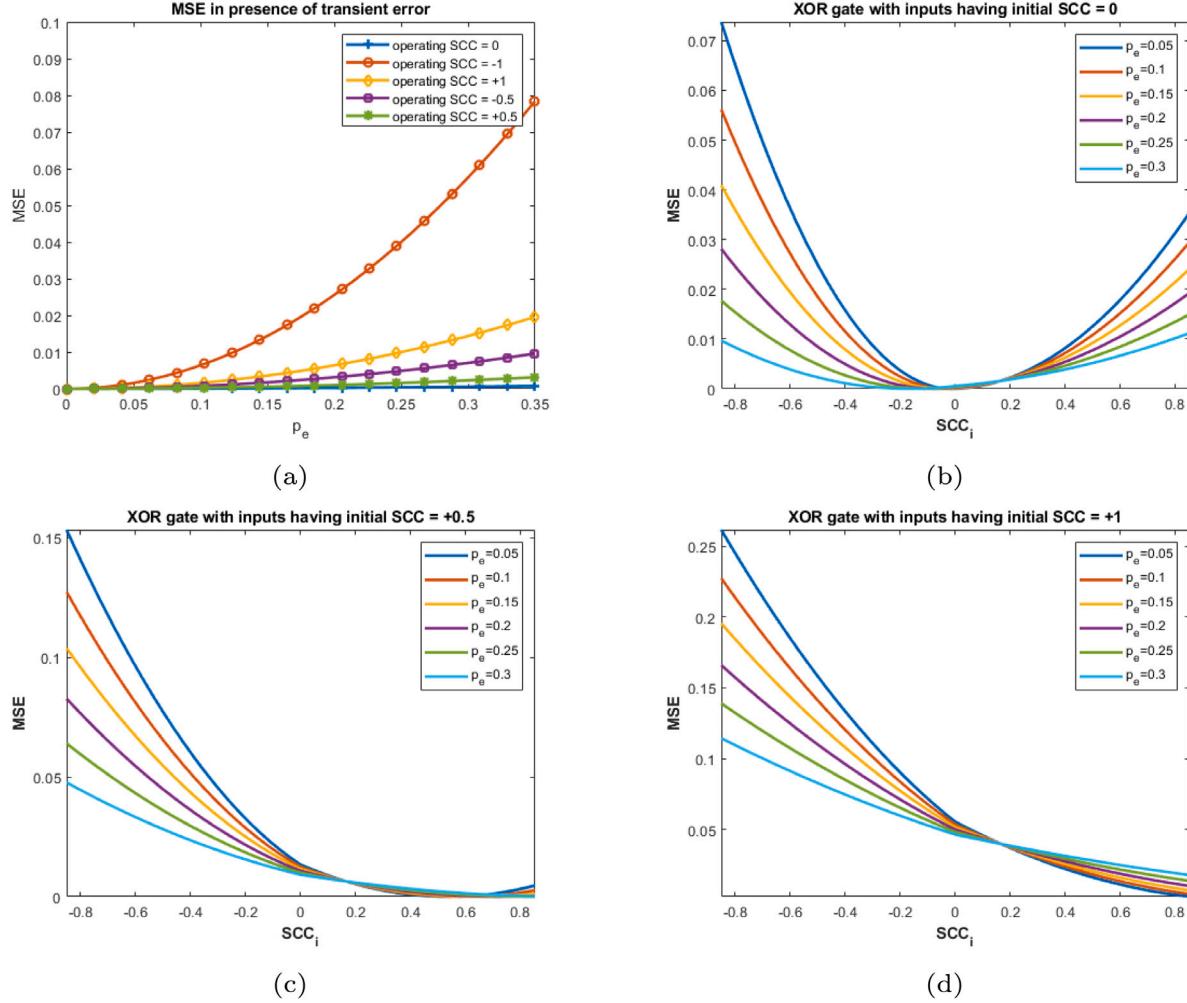


Fig. 2. (a) *MSE* of XOR gate with varying transient errors; Min. *MSE* with *ReCo* for different initial *SCCs* (b) *SCC* = 0 (c) *SCC* = 0.5 (d) *SCC* = 1.

in the probability of a given number when a single level circuit is considered. But for multi-level circuits this can effectively change the overall value. In the next section we focus on two major error sources and study their effect on the behaviour on logic circuits.

3.1. Correlation errors

Correlation between two bitstreams has been identified as a major source of inaccuracy in certain stochastic circuits. Correlation in Stochastic computing indicates that the bitstreams generated by LSFR [19] or SNG [20] inherit some sort of dependence between them (cross-correlation) or between the bits of the same bitstream (auto-correlation). Earlier, correlation in stochastic circuits could only be vaguely identified as inaccurate output caused by a pair of bitstreams when passed through an AND gate. But, recently, correlation in stochastic computing has been quantified and identified with definiteness [20].

To quantify the correlation between input bitstreams X and Y , *SCC* (Stochastic Correlation Coefficient) which is analogous to the similarity coefficient [21] is represented as

$$SCC(X, Y) = \begin{cases} \frac{p_{X \wedge Y} - p_X \cdot p_Y}{\min(p_X, p_Y) - p_X p_Y}, & p_{X \wedge Y} > p_X \cdot p_Y \\ \frac{p_{X \wedge Y} - p_X \cdot p_Y}{p_X p_Y - \max(p_X + p_Y - 1, 0)}, & \text{otherwise} \end{cases}$$

where, $p_{X \wedge Y}$ is obtained by bitwise AND operation between X and Y . Other generalized way of representing the SCC is:

$$SCC(X, Y) = \begin{cases} \frac{n_{11} \cdot n_{00} - n_{01} \cdot n_{10}}{n_{\min}(n_{11} + n_{10}, n_{11} + n_{01}) - (n_{11} + n_{10})(n_{11} + n_{01})}, & n_{11} \cdot n_{00} > n_{01} \cdot n_{10} \\ \frac{n_{11} \cdot n_{00} - n_{01} \cdot n_{10}}{(n_{11} + n_{10})(n_{11} + n_{01}) - n_{\max}(n_{11} - n_{00}, 0)}, & \text{otherwise} \end{cases}$$

where, n_{11} , n_{10} , n_{01} and n_{00} are the respective bit overlaps of X and Y . Thus, the measure of correlation is influenced only by the overlap of similar and dissimilar bits in the bitstreams. Let, $X = 110011110100$, and $Y = 010011110100$, then, $SCC = +1$. But, if $X = 101100010101$, $Y = 11111000101$, $SCC = 0.5$. In this case, not every 1 in Y is influenced by the presence of 1 in that position in X . There is overlapping of 0's in X and 1's in Y as well as 1's in X and 0's in Y . Thus, the pair of bitstreams is positively correlated to a certain degree. Correlation has also been found to have a positive effect on the circuit's behaviour [22,23]. XOR gate acts as an absolute subtractor when inputs are positively correlated, as shown in Fig. 3. Implementing the same function using binary inputs increases hardware complexity [24].

But for boundary values of probability, either, 0 or 1, the measure of *SCC* becomes indeterminate. In both of these cases, it is impossible to change the *SCC* value with the help of any external circuit such as a correlator. To relate to this, consider two SNs $X = 00000000$ and $Y =$

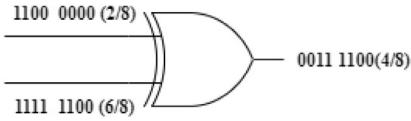


Fig. 3. XOR gate as absolute subtractor when inputs are positively correlated.

11111111. Logic operations on these numbers tend to produce output that will stick to the boundary values itself, either 0 or 1 depending on the SLE. Attempts to change the correlation status will result in a change in probability value, which is undesired. Using a correlator circuit such as [25] will not be able to alter the degree of correlation between X and Y because only grouping of one kind of bit-pair (here 01) will be possible and we lose the leverage of pairing other three bit pairs i.e., 00, 10, 11. For any degree of correlation, we can write the output p_z as a linear combination of its functions at $SCC = 0$ and $SCC = +1$ or -1 [20] given as:

$$p_z = f(p_x, p_y) = (1 + SCC)F_0 - SCC.F_{-1}, \quad \forall SCC < 0 \quad (9)$$

$$p_z = f(p_x, p_y) = (1 - SCC)F_0 + SCC.F_{+1}, \quad \forall SCC > 0 \quad (10)$$

F_0, F_{-1}, F_{+1} are the functions realized by the logic at $SCC = 0, -1, +1$ respectively. Consider an AND gate with inputs p_x and p_y . For different correlation status between inputs, different functions are realized; $F_0 = p_x p_y$ (when inputs are uncorrelated), $F_{+1} = \min(p_x, p_y)$ (positively correlated) and $F_{-1} = \max(p_x + p_y - 1, 0)$ (negatively correlated).

Similarly, functions realized by XOR gate when the inputs are uncorrelated, positively correlated, can be found out as,

$$F_0 = p_x(1 - p_y) + p_y(1 - p_x) = p_x + p_y - 2p_x p_y \quad (11)$$

$$F_1 = |p_x - p_y| \quad (12)$$

3.2. Soft errors

As semiconductor technology advances with reduced feature size and increased scalability, it is becoming more prone to soft errors [6]. The sources of such soft errors have been traced to mainly alpha particles and high energy cosmic rays [26]. Although soft errors are not unique to stochastic circuits, its properties make it more tolerant to soft errors than weighted-binary logic circuits. Soft errors do not affect the circuit physically, but they introduce behavioural changes in the circuit in the form of bit flips by introducing false logic [27,28]. Transient or soft errors are caused due to exposure to external radiation and are greatly increased by manufacturing defects in a chip. These introduce false logic at the output of the circuit [6]. So, if multiple faults strike nodes of a gate, the output may be obtained erroneously. Bit flips are modelled as bit flip error p_e associated with each gate in the circuit as a Bernoulli variable.

Stochastic numbers are analysed as Bernoulli random variables (BRV) represented by their probability of success p_x to perform similar operations as with other BRVs. Errors in BRVs are usually analysed using Mean Square Error (MSE) written as $E_z = E[(p_{ze} - p_z)^2]$, where, p_{ze} and p_z represent the estimated and exact value respectively. A lot of applications involving stochastic circuits are carried out in a noisy environment where the circuit is prone to bit-flip errors. For nano-scale devices, transient or soft errors are growing prominence as the device features are downscaled to sub-micron ranges. The observed output might exceed the error threshold due to the change in the expected value of signals and also due to unwanted correlation introduced during bit flips. For larger circuits, this may be a major concern for accuracy [29].

The presence of soft errors coupled with other inherent error sources may cause model instability in multiple responses. Thus, to achieve the desired level of accuracy irrespective of the environment is a dire

need in this scenario. Soft errors can change the status of correlation between bitstreams that may or may not change the probability value. If an equal number of 1s and 0s are flipped in a bitstream on account of transient faults, then the probability value remains unchanged. However, if the number of bit-changes is unequal that may have an effect on changing the overall probability value. Fig. 4 shows the effect of transient errors on the behaviour of a correlation sensitive stochastic logic elements (SLEs). When fault-free, the AND gate implements multiplication of two numbers. This condition is not true for two other cases, where, p_e at 0.125 hit the input nodes at different bit positions leading to shifted correlation status between two numbers as shown in Fig. 4(b) and (c).

As we increase the transient error the MSE increases exponentially. In case of inputs operating in the negative range of correlation the error surmounts with the incremental injection of soft error rates as shown in Fig. 1(a), 2(a) and 6(a)(red colour). Whereas the same bitstream operating in the positive range of correlation will show reduced MSE with the injection of soft errors (yellow colour). The responses of the correlation-sensitive logic elements like AND, OR and XOR gate with varying transient errors are captured in Fig. 1(a),(b) and (c). These are discussed in detail in the next section, where the motto is to reduce the effect of the transient faults on probabilistic circuits by harnessing some of the unique properties of each of these correlation sensitive circuits.

4. Handling errors in stochastic circuits using the proposed technique

Minimizing errors is crucial since this distorts the output logic level of the circuit. We assume that transient faults at the gates introduced by external factors lead to change in the input as well as output probabilities thereby introducing uncertainty in correlation assumption of the circuit. It is observed that bit flips at different positions due to transient errors may lead to different correlation status between the same bitstreams. Undesired correlation can also lead to different stochastic functions being implemented by the same logic circuit, as shown in Fig. 4 and impedes the natural function to get implemented. Change in correlation status may also result in the change in probability value if an unequal number of 0's and 1's are flipped. Thus, to realize the accurate stochastic function and hence the output in this error scenario a correlation restoration scheme needs to be grabbed, that minimizes the effect of transient error in the circuit.

4.1. The proposed Remodelling Correlation (ReCo) framework

Our work suggests *Remodelling Correlation (ReCo)* technique to cater to the change in the probability assumption at the inputs owing to transient faults. We interpret techniques for correlation-sensitive elements to bring down the MSE to a minimum level. While conducting a study on correlation-sensitive SLEs we demonstrate that every design error can be corrected by introducing correlation to a certain degree at the inputs. We deduce an operating point of the circuit in this incorrect environment with a suitable injection of SCC that reduces MSE to a minimum value. *Algorithm1* searches for a unique solution of the induced correlation within the range $[-1,1]$ to find a minimum error for input parameters. We begin our analysis by considering single SLEs. The flowchart of the proposed framework is shown in Fig. 5.

4.1.1. ReCo analysis for correlation sensitive logic elements with zero correlation assumption

A stochastic circuit implements different real-valued functions when the correlation between input numbers is altered. We assume the target function to be implemented at $SCC(X, Y) = 0$ and any deviation is considered as faulty behaviour of the circuit.

(i) *AND gate*: Consider an AND gate that is inflicted by transient noise. Since we have set $SCC(X, Y) = 0$, we can evaluate $p_z = p_x p_y$ to be true. As we increase the probability of transient error, the observed

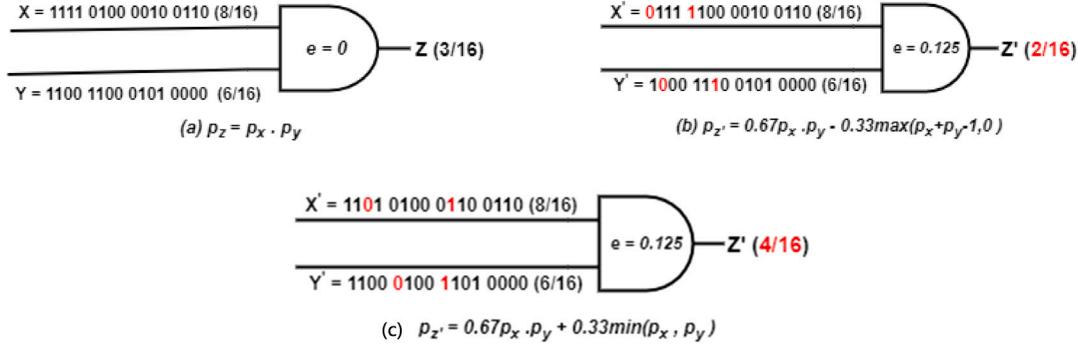


Fig. 4. Bit flips at different positions due to transient errors and its impact on correlation alteration resulting in different Stochastic functions.

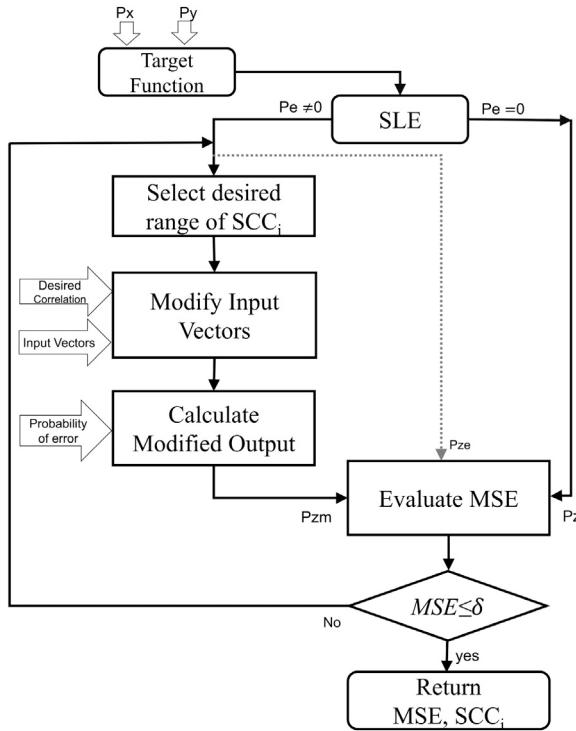


Fig. 5. Flowchart of the proposed framework.

output deviates more from the original output showing an exponential increase in MSE , indicated in Fig. 1(a) (blue). So we employ the proposed method with an aim to reduce the MSE .

We modify each element of the input vector i_{00}, i_{01}, i_{10} , and i_{11} between p_x and p_y to form a new vector consisting of modified elements. For two independent inputs with a zero correlation assumption, we express the input vector as,

$I_0 = [(1 - p_x)(1 - p_y) \quad (1 - p_x)p_y \quad p_x(1 - p_y) \quad p_xp_y]$, assuming $p_x < p_y$ and $p_x + p_y \leq 1$. Let the injected correlation SCC_i be in the range of 0 to +1. For two negatively correlated inputs,

$I_{-1} = [1 - (p_x + p_y) \quad p_y \quad p_x \quad 0]$. The modified input vectors are calculated using Eq. (9)

$$\begin{aligned} i_{00_m} &= (1 + SCC_i)i_{00(I_0)} - SCC_i \cdot i_{00(I_{-1})} \\ &= (1 + SCC_i)(1 - p_x)(1 - p_y) - SCC_i \{1 - (p_x + p_y)\} \end{aligned} \quad (13)$$

$$\begin{aligned} i_{01_m} &= (1 + SCC_i)i_{01(I_0)} - SCC_i \cdot i_{01(I_{-1})} \\ &= (1 + SCC_i)(1 - p_x)p_y - SCC_i p_y \end{aligned} \quad (14)$$

$$i_{10_m} = (1 + SCC_i)i_{10(I_0)} - SCC_i \cdot i_{10(I_{-1})}$$

$$= (1 + SCC_i)(1 - p_y)p_x - SCC_i p_x \quad (15)$$

$$\begin{aligned} i_{11_m} &= (1 + SCC_i)i_{11(I_0)} - SCC_i \cdot i_{11(I_{-1})} \\ &= (1 + SCC_i)p_x p_y \end{aligned} \quad (16)$$

where, $i_{00(I_0)}, i_{01(I_0)}, i_{10(I_0)}, i_{11(I_0)}$ are four elements of input vector I_0 . Similarly, $i_{00(I_{-1})}, i_{01(I_{-1})}, i_{10(I_{-1})}$, and $i_{11(I_{-1})}$ are four elements of vector I_{-1} . Now, the modified vectors of I_{SCC_m} can be represented as $I_{SCC_m} = [i_{00_m} \quad i_{01_m} \quad i_{10_m} \quad i_{11_m}]$. Reducing Eqs. (13)–(16) further we get,

$$I_{SCC_m} = \begin{bmatrix} 1 - (p_x + p_y) + p_x p_y (1 + SCC_i) \\ -p_y (p_x + p_y SCC_i - 1) \\ -p_x (p_y + p_y SCC_i - 1) \\ p_x p_y (SCC_i + 1) \end{bmatrix}^T \quad (17)$$

Similarly, for $p_x + p_y > 1$,

$$I_{SCC_m} = \begin{bmatrix} -(p_y - 1)(p_x SCC_i - p_x + 1) \\ p_y (SCC_i - 1)(p_x - 1) - SCC_i (p_x - p_y) \\ p_x (SCC_i - 1)(p_y - 1) \\ p_x SCC_i - p_x p_y (SCC_i - 1) \end{bmatrix}^T \quad (18)$$

Algorithm 1: ReCo analysis for a single gate

```

1: Input  $p_x, p_y, p_e, input\_Gate$ ; Output  $MSE_i, SCC_i$ 
2: ReCo(input_Gate)
3:  $[p_x, p_y] =$  input probabilities of input_Gate;
4:  $p_e =$  probabilities of transient errors of input_Gate;
5:  $True\_Output = Eval(p_x, p_y, SCC)$ ;
6: for  $SCC_i = -1; SCC_i \leq +1; SCC_i + = 0.001$  do
    $Modified\_Output = Eval(p_x, p_y, p_e, SCC_i);$ 
    $MSE_i = True\_Output - Modified\_Output$ 
   if  $MSE_i \leq \delta$  then
      | return  $MSE_i, SCC_i$ 
   end if
   end for
7: return  $\text{argmin}_{MSE_i} \{MSE_i, SCC_i\}$ 
```

For an AND gate with a given error rate p_e , the modified output p_{zm} as a function of I_{SCC_m} can be written as,

$$p_{zm} = I_{SCC_m} \times \begin{bmatrix} 1 - p_e & p_e \\ 1 - p_e & p_e \\ 1 - p_e & p_e \\ p_e & 1 - p_e \end{bmatrix} \quad (19)$$

Thus, p_{zm} is observed as $f(SCC_i)$. We try to make p_{zm} close to p_z to reduce the observed error p_{ze} . Thus,

$$p_{zm} = p_e + p_x p_y (1 + SCC_i) (1 - 2p_e) \quad (20)$$

The MSE which is $(p_{zm} - p_z)^2$ is calculated as,

$$MSE_{and} = \{p_e + p_x p_y (SCC_i - 2p_e - 2p_e SCC_i)\}^2 \quad (21)$$

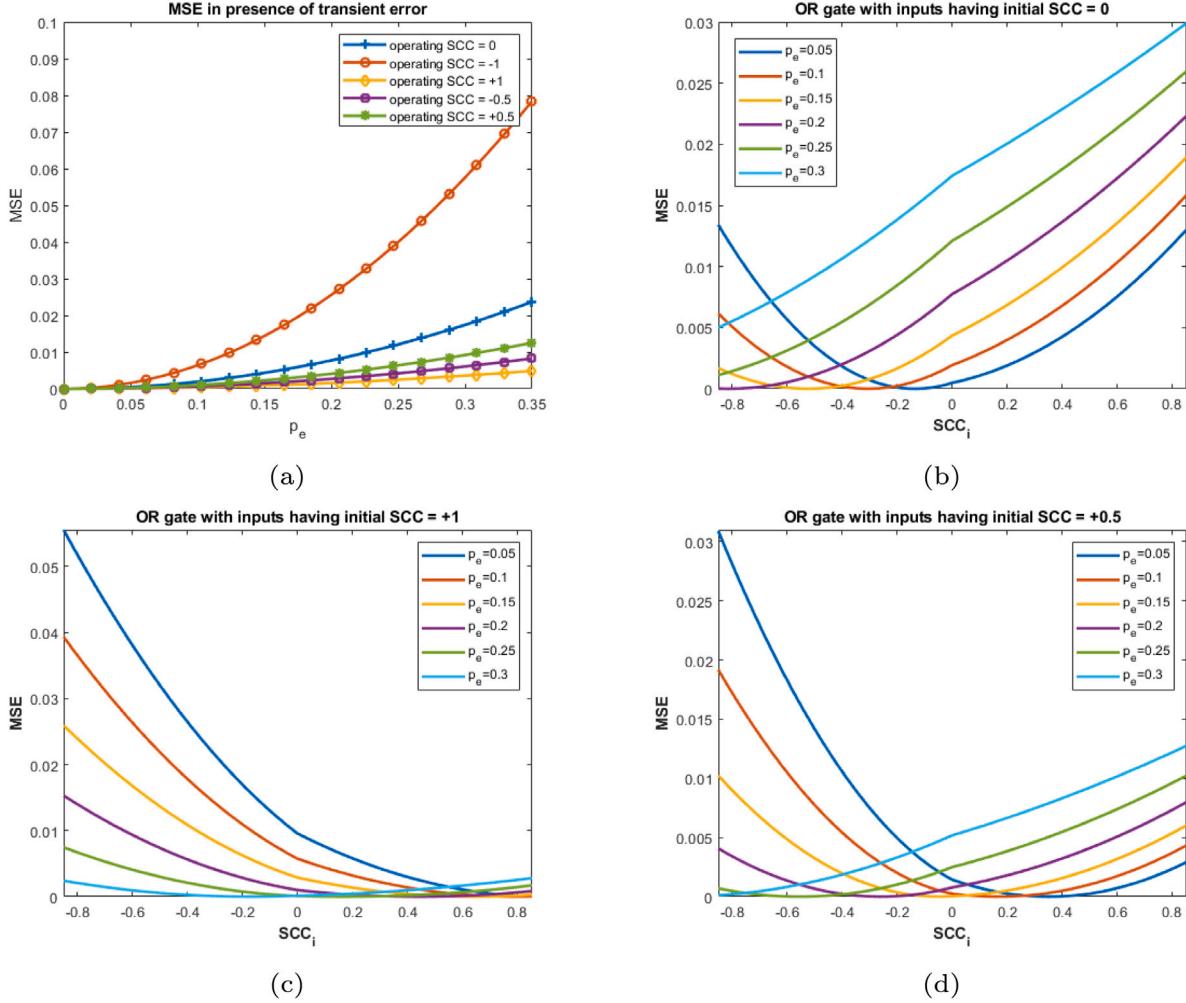


Fig. 6. (a) MSE of OR gate with varying transient errors; Min. MSE obtained using *ReCo* for different initial $SCCs$ (b) $SCC = 0$ (c) $SCC = 1$ (d) $SCC = 0.5$.

The induced SCC_i which reduces MSE to a minimum possible value within the range $[-1, +1]$ is obtained by differentiating Eq. (21) w.r.t SCC and equate it to 0.

$$2p_x p_y (1 - 2p_e) \{p_e + p_x p_y (SCC_i - 2p_e - 2p_e SCC_i)\} = 0$$

$$\therefore SCC_i = \frac{-(p_e - 2p_e p_x p_y)}{p_x p_y (1 - 2p_e)} \quad (22)$$

Eq. (22) dictates the condition of reaching a minimum value of MSE for SCC_i in the range $[-1, 0]$. Similarly, when $p_x + p_y > 1$ with $p_x < p_y$, SCC_i can be evaluated as,

$$SCC_i = \frac{(p_e - p_x - p_y + p_x p_y - 2p_e p_x p_y + 1)}{(2p_e - 1)(p_x - 1)(p_y - 1)} \quad (23)$$

where, $p_x, p_y > 0$ and $p_x < p_y$. Thus, $p_x p_y > 0$. Expressions are derived assuming negative induction of correlation. However, nothing in the derivation prevents SCC_i from being positive to achieve the minimum MSE .

To exploit the simplicity of equations and to achieve the maximum possible accuracy in calculations, parameters appearing in equations are verified graphically. Fig. 1(b) shows different values of induced correlation to obtain zero error at the output at different error rates. Note that, Eqs. (22), (23) always hold for $p_e < 0.5$. Using similar analysis, we arrive at different sets of equations for $p_x > p_y$.

Example 1. Consider an AND gate with $p_x = 0.3$ and $p_y = 0.6$. The error-free output is $p_z = 0.3 \times 0.6 = 0.18$. The observed output is

$p_{ze} = 0.28$ at $p_e = 0.15$. Thus,

$$I_{SCC_m} = \begin{bmatrix} (0.28 + 0.18SCC_i) \\ (0.42 - 0.18SCC_i) \\ (0.12 - 0.12SCC_i) \\ (0.18 + 0.12SCC_i) \end{bmatrix}$$

Thus, $p_{zm} = 0.18SCC_i + 0.64p_e - 0.36p_e SCC_i + 0.18$ and $MSE = \frac{(9SCC_i + 32p_e - 18p_e SCC_i)^2}{40000}$. Error reduces to 0 for $SCC_i = -0.76$. It is observed that MSE can be reduced to 0 if $p_e \leq 0.2$ (a considerate limit). The results are confirmed graphically considering different values of p_x and p_y at $p_e = 0.125$ as shown in Fig. 7(a).

(ii) **XOR gate:** At $SCC = 0$, the XOR gate implements $p_z = p_x(1 - p_y) + p_y(1 - p_x)$. Any deviation from the target function on account of transient error is considered as a contribution to MSE . A similar foregoing approach is adopted in the analysis of the XOR gate to operate at a minimum MSE under different transient error rates. Now, p_{zm} for $p_x < p_y$ is written as,

$$p_{zm} = I_{SCC_m} \times \begin{bmatrix} 1 - p_e & p_e \\ p_e & 1 - p_e \\ p_e & 1 - p_e \\ 1 - p_e & p_e \end{bmatrix} \quad (24)$$

Substituting I_{SCC_m} from Eq. (17),

$$p_{zm} = p_e - p_x p_y - 2p_e(p_x + p_y - 2p_x p_y) - 2p_x p_y SCC_i(1 - 2p_e)$$

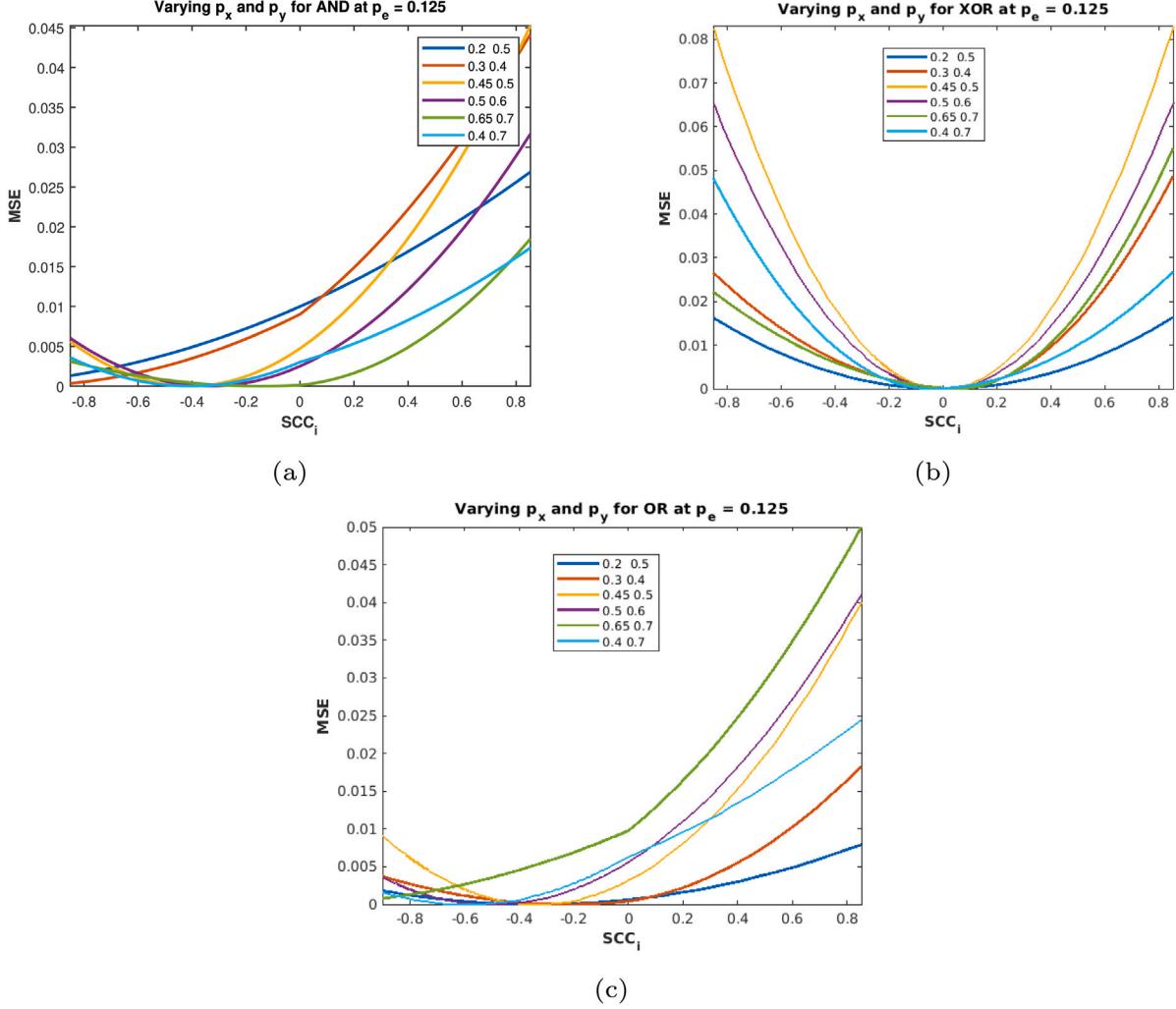


Fig. 7. MSE after ReCo analysis of correlation sensitive gates with $p_e = 0.125$ subjected to transient error 0.125 for different input probabilities; (a) AND gate (b) XOR gate (c) OR gate.

$$MSE_{xor} = \{2p_e(p_x + p_y) - p_e + 2p_x p_y (SCC_i - 2p_e - 2p_e SCC_i)\}^2 \quad (25)$$

We differentiate Eq. (25) w.r.t SCC_i and equate it to 0.

$$\begin{aligned} & -2(2p_x p_y - 4p_e p_x p_y)(p_e - 2p_e p_x - 2p_e p_y - \\ & 2p_x p_y (SCC_i + 2p_e + 2p_e SCC_i)) = 0 \end{aligned}$$

$$\therefore SCC_i = \frac{p_e - 2p_e(p_x + p_y) + 4p_e p_x p_y}{2p_x p_y - 4p_e p_x p_y} \quad (26)$$

Similarly, for $p_x + p_y > 1$,

$$SCC_i = \frac{p_e - 2p_e p_x - 2p_e p_y + 4p_e p_x p_y}{4p_e + 2(p_x + p_y)(1 - 2p_e) - 2p_x p_y(1 + 2p_e) - 2} \quad (27)$$

Fig. 2(b) shows different values of SCC_i to reach zero MSE at different values of p_e , which is consistent with Eq. (26).

Example 2. Consider XOR gate with inputs $p_x = 0.3$ and $p_y = 0.6$. Thus $p_z = 0.54$ and $p_{ze} = 0.52$ at $p_e = 0.15$. By substituting I_{SCC_m} we find $p_{zm} = 0.72p_e - 0.08p_e - 0.36SCC_i + 0.54$ and hence $MSE = \frac{(9SCC_i + 2p_e - 18p_e SCC_i)^2}{625}$. Thus, MSE reduces to 0 at $SCC_i = 0$.

From the experiment (see Fig. 2(a)) it is inferred that the XOR gate is least responsive to probabilistic errors and hence the smallest contributor to the overall MSE in the circuit. Also, it is evident from

Fig. 2 (b) that the XOR gate is most sensitive to changes in correlation. Thus for $0 \leq p_e \leq 0.3$, the error can be reduced to 0 in contrary to other gates, AND and OR (see Figs. 1(b) and 6(b)). This is also validated using different values of p_x and p_y at $p_e = 0.125$ which is shown in Fig. 7(b).

(iii) OR gate: For uncorrelated numbers, OR gate implements $p_z = p_x + p_y - p_x p_y$. In presence of transient error let the function be p_{ze} . We eliminate this error by introducing the ReCo block at inputs to inject the desired correlation. The modified output p_{zm} is then calculated using Eq. (17).

$$\begin{aligned} p_{zm} &= I_{SCC_m} \times \begin{bmatrix} 1 - p_e & p_e \\ p_e & 1 - p_e \\ p_e & 1 - p_e \\ p_e & 1 - p_e \end{bmatrix} \\ \therefore p_{zm} &= p_e + p_x + p_y - 2p_e p_x - 2p_e p_y - p_x p_y - \\ & p_x p_y SCC_i + 2p_e p_x p_y + 2p_e p_x p_y SCC_i \end{aligned}$$

$$\therefore MSE_{or} = (p_e \{1 + p_x p_y\} - 2p_e \{p_x + p_y\} - p_x p_y SCC_i \{1 - 2p_e\})^2 \quad (28)$$

We differentiate Eq. (28) w.r.t SCC_i and equate it to 0.

$$\begin{aligned} & -2(p_x p_y - 2p_e p_x p_y)(p_e - 2p_e p_x - 2p_e p_y - \\ & -p_x p_y SCC_i + 2p_e p_x p_y + 2p_e p_x p_y SCC_i) = 0 \end{aligned}$$

$$\therefore SCC_i = \frac{p_e - 2p_e(p_x + p_y) + 2p_e p_x p_y}{p_x p_y - 2p_e p_x p_y} \quad (29)$$

Similarly for $p_x + p_y > 1$,

$$SCC_i = \frac{2p_e(p_y - p_x) - p_e + p_x p_y(1 - 2p_e)}{p_x p_y(1 - 2p_e)} \quad (30)$$

Fig. 6(b) shows different values of induced correlation to obtain zero error at the output of OR gate for different values of p_e .

Example 3. Consider an OR gate with inputs $p_x = 0.3$ and $p_y = 0.6$ at $p_e = 0.15$. The error free output $p_z = 0.72$ and the observed output $p_{ze} = 0.654$. Using Eq. 2 $p_{zm} = 0.36p_e SCC_i - 0.44p_e - 0.18SCC_i + 0.72$ and $MSE = \frac{(9SCC_i + 22p_e - 18p_e SCC_i - 6)^2}{2500}$. Thus, MSE can be reduced to zero at $SCC_i = -0.5238$.

Thus, similar to the AND gate, the MSE of the OR gate can be reduced to zero if the transient error is below a certain limit, say 0.2, although the reduction is less than that of the AND gate.

From the analysis, it is observed that the OR gate is least sensitive to changes in correlation, whereas XOR is the highest. AND gate is intermediate to them. It is also identified that the XOR gate is least affected by transient error, whereas AND gate is mostly influenced by the presence of transient error. So MSE increases immensely when the error is imposed on an AND gate. These properties of the XOR gate make it a suitable choice for the analysis of an error-resilient circuit design. In the next section, this idea is implemented on complex circuits that focus to minimize MSE with minimum hardware using the proposed methodology.

4.1.2. ReCo analysis for correlation-sensitive logic elements with non-zero correlation assumption

Those SLEs which are sensitive to correlation implement an altogether different stochastic function. An example is shown with the help of an AND gate in **Fig. 4(b), (c)**. In this section, we reconsider SLEs with transient errors having an apriori correlation assumption. We invoke ReCo analysis to suppress MSE and formulate the underlying conditions in support of that. Two distinct cases of initial correlation assumption are discussed, i.e., $SCC = +0.5$ and $SCC = +1$ and perform a similar analysis to reduce errors at different degrees of transient faults. The target function is obtained considering an initial non-zero and positive value of correlation. It is observed that for an existing negative SCC between input variables the effect of transient errors in the circuit element is enhanced. Thus, such cases are excluded in our analysis.

(i) **AND gate:** The analysis begins by setting a non-zero and positive initial correlation between p_x and p_y . The intersection of p_e with the previously set positive value of correlation between inputs implicitly assumes that there is a shift in the value of correlation to arrive at the minimum MSE .

(a) **With existing $SCC(X, Y) = 1$:** For positively correlated numbers, AND gate implements $p_z = \min(p_x, p_y)$. We counter the effect of transient error on the circuit by introducing a desired SCC_i obtained using following derivations.

$$I_{SCC_m(+1)} = \begin{bmatrix} -(p_y - 1)(p_x SCC_i - p_x + 1) \\ p_y(SCC_i - 1)(p_x - 1) - SCC_i(p_x - p_y) \\ p_x(SCC_i - 1)(p_y - 1) \\ p_x - p_x p_y SCC_i (SCC_i - 1) \end{bmatrix} \quad (31)$$

The modified output p_{zm} is calculated as

$$p_{zm} = I_{SCC_m(+1)} \times M_{and} = p_e p_x - p_e(p_x + p_y - 1) + p_e p_y \quad (32)$$

For $p_x + p_y \leq 1$, modified MSE is,

$$MSE_{and} = (p_e - p_x + p_x(1 - 2p_e)\{SCC_i(1 - p_y) + p_y\})^2 \quad (33)$$

Differentiating Eq. (33) w.r.t SCC_i and equate it to 0,

$$SCC_i = \frac{p_x - p_e - p_x p_y(1 - 2p_e)}{p_x(2p_e - 1)(p_y - 1)} \quad (34)$$

$$SCC_i = \frac{(p_x - p_e + 2p_e p_x + p_x p_y)}{p_x - 2p_e p_x - p_x p_y + 2p_e p_x p_y}, p_x + p_y > 1 \quad (35)$$

(b) **Any positive intermediate correlation, $SCC(X, Y) = 0.5$:** Now consider any intermediate positive correlation between the numbers, say +0.5. The function implemented by AND logic at $SCC = 0.5$ is $p_z = 0.5p_x(1 + p_y)$ for $p_x < p_y$. In presence of transient error in the circuit, we modify input vectors as I_{SCC_m} ,

$$I_{SCC_m(+0.5)} = \begin{bmatrix} (1 - p_y)(0.5p_x SCC_i - p_x + 1) \\ 0.5p_y(1 - p_x) - SCC_i(0.5p_x + 0.5p_y - p_x p_y) \\ p_x(SCC_i - 1)(p_y - 1) \\ p_x \cdot (p_y + 0.5SCC_i - 0.5p_y SCC_i) \end{bmatrix} \quad (36)$$

$$p_{zm} = I_{SCC_m(+0.5)} \times M_{and} = 0.5SCC_i p_x (p_y - p_e + p_e p_y) + SCC_i p_x (0.5 - p_e + p_e p_y) + p_x p_y (1 - 2p_e) + p_e \quad (36)$$

$$MSE_{and} = p_e(1 - 2p_x p_y) - p_x(1 - p_y) \\ (0.5 + p_e SCC_i) + 0.5p_x(1 - p_y)(1 - p_e) SCC_i \quad (37)$$

Differentiating Eq. (37) w.r.t SCC_i and putting it to 0, gives

$$SCC_i = \frac{0.5p_x(1 - p_y) + 2p_e p_x p_y - p_e}{0.5p_x(1 - p_y)(1 - 3p_e)} \quad (38)$$

$$SCC_i = \frac{2p_e p_x - p_x - 2p_e + p_x p_y + 2p_e p_x p_y}{2p_x - 4p_e p_x - 2p_x p_y + 4p_e p_x p_y}, p_x + p_y > 1. \quad (39)$$

Example 4. Consider $p_x = 0.3$, $p_y = 0.6$ and $p_e = 0.15$. From Eq. (36), $p_{zm} = 0.1SCC_i + 0.64p_e - 0.24p_e SCC_i + 0.18$. We invoke Eq. (37) to obtain $MSE = 1.6 \times 10^{-3}(3SCC_i + 16p_e - 6p_e SCC_i - 3)^2$. Thus, for $p_e = 0.15$, MSE can be reduced to zero by injecting $SCC_i = +0.2857$.

With $SCC = +0.5$ between inputs, p_z and p_{ze} are 0.24 and 0.32. Using Eq. (37), $p_{zm} = 0.1SCC_i + 0.64p_e - 0.24p_e SCC_i + 0.18$. Thus, $MSE = \frac{(3SCC_i + 32p_e - 9p_e SCC_i - 3)^2}{40000}$. Thus unlike the previous case, MSE can be reduced to zero only for $p_e < 0.15$ by injecting suitable positive SCC_i .

(ii) **XOR gate:** We assume a positive definite correlation between the inputs of an XOR gate and using similar analysis we derive from the condition for minimum MSE .

(a) **With existing $SCC(X, Y) = 1$:** With positively correlated inputs, the XOR gate implements $p_z = F_{+1} = |p_x - p_y|$. The deviation from this assumption under the error scenarios can be encountered by finding a suitable operating point of the circuit by defining SCC_i using the following derivations.

We modify the output by introducing the desired correlation such that,

$$p_{zm} = p_e + p_x + p_y - 2SCC_i p_x - 2p_e p_x - 2p_e p_y - 2p_x p_y + 4p_e p_x SCC_i + 2p_x p_y SCC_i + 4p_e p_x p_y - 4p_e p_x p_y SCC_i \quad (40)$$

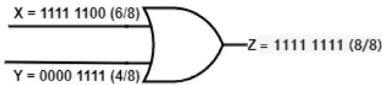
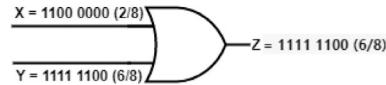
$$MSE_{xor} = (p_e + 2p_x - 2p_e(p_x + p_y) - 2p_x(1 - 2p_e)\{SCC_i(1 + p_y) + p_y\})^2 \quad (41)$$

Differentiating Eq. (41) w.r.t SCC_i and equating it to 0,

$$SCC_i = \frac{p_e(1 - 2p_x)(1 - 2p_y) + 2p_x(1 - p_y)}{2p_x(2p_e - 1)(p_y - 1)} \quad (42)$$

$$SCC_i = \frac{p_e - 2p_x + 2p_e p_x - 2p_e p_y + 2p_x p_y}{2p_x - 4p_e p_x - 2p_x p_y + 4p_e p_x p_y} \quad (43)$$

Example 5. Consider $p_x = 0.3$, $p_y = 0.6$, $p_e = 0.15$. Thus, $p_z = 0.3$ and $p_{ze} = 0.36$ and $p_{zm} = 0.48p_e SCC_i - 0.0800p_e - 0.24SCC_i + 0.54$. Now, $MSE = \frac{(9SCC_i + 2p_e - 18p_e SCC_i - 6)^2}{625}$, which can be reduced to zero when $SCC_i = 0.933$.

(a) $F_{-1} = \min(p_x + p_y, 1)$ (b) $F_{+1} = \max(p_x, p_y)$ **Fig. 8.** OR gate implementing different functions when inputs have different correlation status.

(b) Any positive intermediate correlation, $\text{SCC}(X, Y) = 0.5$: In this case, the resultant function is $p_z = F_{+0.5} = -p_y(p_x - 1)$ when $p_x < p_y$. The error-free output is obtained using modified output defined by input vectors in $I_{\text{SCC}_{m(0.5)}}$.

$$\begin{aligned} p_{zm} &= p_e + p_x + p_y - 5\text{SCC}_i p_x - 2p_e p_x - 2p_e p_y - 2p_x p_y \\ &+ 10p_e p_x \text{SCC}_i + 5\text{SCC}_i p_x p_y + 4p_e p_x p_y - 10p_e p_x p_y \text{SCC}_i \\ \text{MSE}_{xor} &= 0.25(2p_e + 2p_x - 3p_x \text{SCC}_i - 4p_e p_x - 4p_e p_y - 2p_x p_y \\ &+ 5p_e p_x \text{SCC}_i + 3p_x p_y \text{SCC}_i + 8p_e p_x p_y - 5p_e p_x p_y \text{SCC}_i)^2 \end{aligned} \quad (44)$$

Differentiating Eq. (44) w.r.t SCC_i and equate it to 0,

$$\text{SCC}_i = \frac{2p_e(1 - 2p_y)(1 - 2p_x) + 2p_x(1 - p_y)}{p_x(5p_e - 3)(p_y - 1)}, p_x + p_y \leq 1 \quad (45)$$

$$\text{SCC}_i = \frac{p_e - p_x - 2p_e p_y + p_x p_y + 2p_e p_x p_y}{2p_x - 4p_e p_x - 2p_x p_y + 4p_e p_x p_y}, p_x + p_y > 1 \quad (46)$$

With positively correlated inputs the induced SCC_i can generally be written in the form,

$$\text{SCC}_i = \frac{p_e(1 - 2p_x)(1 - 2p_y) + 2\text{SCC} p_x(1 - p_y)}{-p_x(p_y - 1)(\text{SCC} - p_e - 3\text{SCC} p_e + 1)} \quad (47)$$

Example 6. Let $p_x = 0.3$, $p_y = 0.6$ and $p_e = 0.15$. Thus, $p_z = 0.42$ and $p_{ze} = 0.444$. And $p_{zm} = 1.12p_e \text{SCC} - 0.08p_e - 0.56\text{SCC} + 0.54$. Thus, $\text{MSE} = 0.0016(14\text{SCC}_i + 2p_e - 28p_e \text{SCC}_i - 3)^2$, which can be reduced to zero when $\text{SCC}_i = +0.275$.

(iii) OR gate: Consider an OR gate with inputs that are positively correlated. Based on different correlation status, OR gate implements different stochastic functions as shown in Fig. 8. Using ReCo, we attempt to derive the condition for attaining the smallest MSE under transient error scenarios.

(a) With existing $\text{SCC}(X, Y) = 1$: The OR gate implements $p_z = \max(p_x, p_y)$ when two numbers are positively correlated. We can similarly find the operating SCC_i to obtain a minimum MSE under error scenarios.

$$\begin{aligned} \therefore p_{zm} &= p_e + p_y - p_e p_x - 2p_e p_y + p_e p_x \text{SCC}_i + p_e p_x p_y \\ &- p_e p_x p_y \text{SCC}_i - p_x(\text{SCC}_i - 1)(p_e - 1)(p_y - 1) \end{aligned} \quad (48)$$

$$\begin{aligned} \text{MSE}_{or} &= (p_e + p_x - p_x \text{SCC}_i - 2p_e p_x - 2p_e p_y - p_x p_y \\ &+ 2p_e p_x \text{SCC}_i + p_x p_y \text{SCC}_i + 2p_e p_x p_y - 2p_e p_x p_y \text{SCC}_i)^2 \end{aligned} \quad (49)$$

$$\text{SCC}_i = \frac{p_e + p_x - 2p_e p_x - 2p_e p_y - p_x p_y + 2p_e p_x p_y}{p_x(2p_e - 1)(p_y - 1)} \quad (50)$$

$$\therefore \text{SCC}_i = \frac{p_e - p_x - 2p_e p_y + p_x p_y}{p_x - 2p_e p_x - p_x p_y + 2p_e p_x p_y} \quad (50)$$

Example 7. Consider $p_x = 0.3$ and $p_y = 0.6$, $p_z = 0.6$ and $p_{ze} = 0.57$ for $p_e = 0.15$. $p_{zm} = 0.24p_e \text{SCC} - 0.44p_e - 0.12\text{SCC} + 0.72$ with the help of above equations (70) and (71), $\text{MSE} = \frac{(9\text{SCC} + 22p_e - 18p_e \text{SCC} - 6)^2}{40000}$. Thus, MSE can be reduced to zero for $p_e = 0.15$ when $\text{SCC}_i = +0.432$.

(b) Any positive intermediate correlation: If we consider any positive intermediate correlation, such as, $\text{SCC}(X, Y) = +0.5$, the stochastic function realized by OR gate is given by; $p_z = 0.5p_x + p_y - 0.5p_x p_y$.

To determine the induced SCC_i that minimizes MSE under specified error scenarios, the following derivations will suffice.

$$\begin{aligned} p_{zm} &= p_e + p_y - p_e p_x - 2p_e p_y + 0.5p_e p_x \text{SCC}_i + p_e p_x p_y \\ &0.5p_e p_x p_y \text{SCC}_i - p_x(\text{SCC}_i - 1)(p_e - 1)(p_y - 1) \end{aligned} \quad (51)$$

Now, MSE in this case is calculated as;

$$\begin{aligned} \text{MSE}_{or} &= 0.25(2p_e + p_x - 2p_x \text{SCC}_i - 4p_e p_x - 4p_e p_y - p_x p_y \\ &+ 3p_e p_x \text{SCC}_i + 2p_x p_y \text{SCC}_i + 4p_e p_x p_y - 3p_e p_x p_y \text{SCC}_i)^2 \end{aligned} \quad (52)$$

$$\text{SCC}_i = \frac{2p_e + p_x - 4p_e p_x - 4p_e p_y - p_x p_y + 4p_e p_x p_y}{(2p_x - 3p_e p_x - 2p_x p_y + 3p_e p_x p_y)} \quad (53)$$

Similarly, for $p_x + p_y > 1$

$$\text{SCC}_i = \frac{2p_e - p_x - 2p_e p_x - 4p_e p_y + p_x p_y + 2p_e p_x p_y}{2p_x - 4p_e p_x - 2p_x p_y + 4p_e p_x p_y} \quad (54)$$

Example 8. Consider $p_x = 0.3$, $p_y = 0.6$ and $p_e = 0.15$. Thus, $p_z = 0.66$ and $p_{ze} = 0.612$. $p_{zm} = 0.24p_e \text{SCC} - 0.44p_e - 0.12\text{SCC} + 0.72$. Using Eq. (52), $\text{MSE} = \frac{(9\text{SCC} + 22p_e - 18p_e \text{SCC} - 3)^2}{40000}$. Thus, an error-free output can be obtained when the injected correlation is -0.05 .

Thus for an OR gate, induced correlation are mostly obtained in the positive range when the initial correlation between the numbers is +1 as shown in Fig. 6(c), while for any intermediate correlation existing between numbers, the injected SCC_i values are predominantly in the negative range except for $p_e \leq 0.1$. The simulation results are given in Fig. 6(d). For any positive correlation SCC , the expressions can be generalized as:

$$\text{SCC}_i = \frac{p_e + \text{SCC} p_x(1 - p_y) - 2p_e(p_x + p_y) + 2p_e p_x p_y}{p_x(1 - p_y)(1 - p_e) - \text{SCC} \cdot p_e \cdot p_x(1 - p_y)} \quad (55)$$

4.2. The proposed error detector circuit

Consider an SLE operating in a noisy environment. The system level representation of the error correction mechanism for such an SLE is shown in Fig. 10. The inputs to the unit are p_x and p_y , error p_e and output is the desired value p_{zm} . The control circuit or the error detector circuit is used to determine the amount of deviation of an erroneous output from an error-free output. It comprises a subtractor, a squarer and a comparator which determines the amount of error that is to be reduced. Auxiliary circuits like the shuffle buffer and the synchronizer are used to adjust the correlation between the input bitstreams. The output of the control unit is fed to the ReCo block consisting of a correlator circuit that generates the desired SCC_i to minimize MSE. The control circuit is described below.

4.2.1. Synchronizer

The synchronizer [25] is a finite state machine that pairs up a maximum number of input bits to 00 or 11, restoring their respective probabilities. This unit is placed between two uncorrelated sequences, p_{ze} and p_z to introduce positive correlation between sequences p_{ze_s} and p_{z_s} . If the bits in p_{ze} and p_z are equal, then the corresponding bits are given as output. When bits are dissimilar, depending upon input values 1 or 0, if they are in state $S1$, are changed to $S0$ and $S2$, both 0 or both 1 are given as output. In this process, the probabilities of p_{ze} and p_z are kept unchanged.

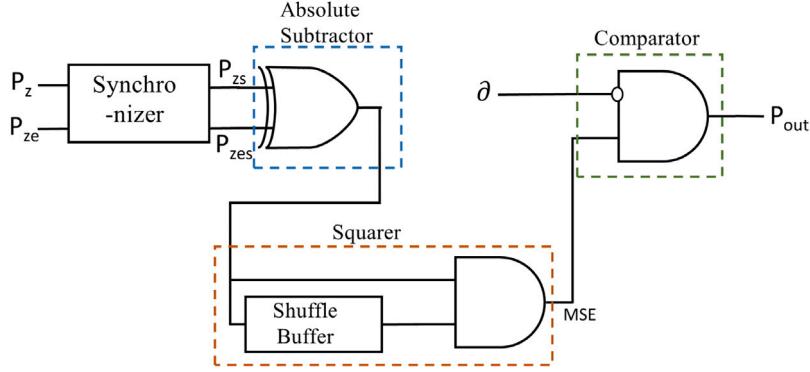


Fig. 9. The error detector circuit..

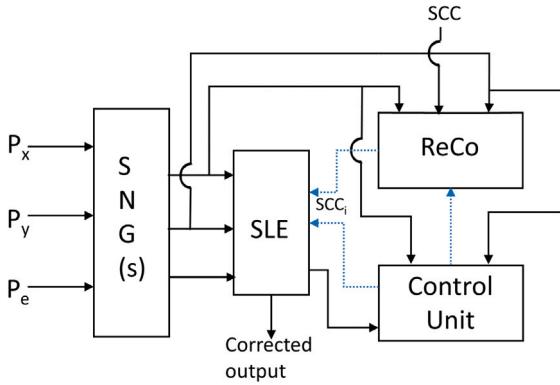


Fig. 10. System-Level representation of Error Correction mechanism.

4.2.2. Subtractor

The difference between the error value p_{ze} and the actual value p_z is calculated to check the amount of deviation. The XOR gate performs absolute subtraction i.e., $p_{diff} = |p_{z_s} - p_{ze_s}|$, when p_{z_s} and p_{ze_s} are positively correlated [20] which is done using a synchronizer.

4.2.3. Squarer

Multiplication of two uncorrelated SNs is performed by an AND gate, but fails to implement the squaring operation [20] when the same input sequence is given. When p_{diff} is squared to obtain the MSE, it is necessary to minimize the correlation between SNs. A shuffle buffer circuit is used [25] to reduce the correlation between inputs to obtain the accurate squaring operation. It includes a multiplexer, three D Flip-flops and a Random Number Generator to generate numbers between 0 and 1 [25].

4.2.4. Comparator

The stochastic comparator, which is shown in Fig. 9 compares the obtained MSE with a very small value, say, δ (0.0001). When two correlated inputs are given to an AND gate with an inverter to one of its inputs the stochastic function [20] is given by $P_{out} = \max((MSE - \delta), 0)$. When MSE which is representative of the error in computation is lesser than the error-tolerance of the circuit δ , a bitstream of 0's is obtained at the output of the comparator which indicates that the output is obtained satisfactorily.

5. Formalization of ReCo technique for stochastic circuits

The next step is to formally apply the proposed technique in a combinational circuit. For a two-input multilevel circuit as shown in Fig. 11, we use circuit PTM obtained as, $M_{ckt} = (I \otimes M_{and} \otimes I).(I \otimes F_2 \otimes I).(M_{or} \otimes M_{or}).M_{and}$.

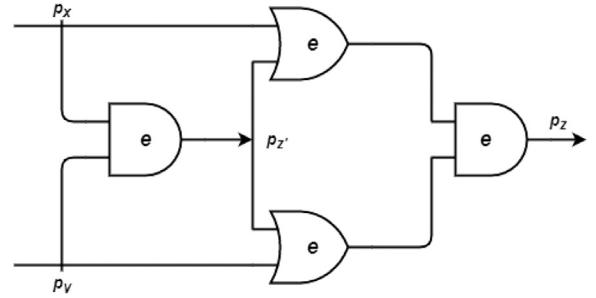


Fig. 11. A 2 input multi-level circuit.

Consider $p_x = 0.3$ and $p_y = 0.6$. The error free output is $p_z = 0.18$. If $p_e = 0.125$, the observed output, $p_{ze} = 0.33$. Using the proposed method, $MSE = (0.066SCC + 0.0388)^2$ which shows that MSE can be reduced to zero for $SCC \approx -0.58$.

We will now discuss two distinct cases of fault correction in multi-input multi-level circuits.

5.1. ReCo I: Error minimization for Multiple-input Single output (MISO) stochastic logic circuits

Let us consider correlation-sensitive blocks interconnected in a fashion as shown in Fig. 12. The quantification of SCC is only available for two signals in literature, so we consider two input gates in the circuit model. The block diagram consists of $i = 1, 2, \dots, n$ levels and each level consists of multiple two input gates. Consider that one or more gates at different levels are subject to transient faults which result in an increased MSE at the output. The MSE is minimized by selecting suitable candidates for $ReCo$ analysis using the proposed Algorithm 2 which is discussed below.

Definition 2. An observed error at the primary output(s) driven by one or more gates, can be minimized by suitably injecting correlation at the prioritized input gates defined by one or more faulty gates in the error path P.

Assume that the correlation between input pairs for gates at level 1 as $SCC_{11}, SCC_{12}, SCC_{13}, \dots, SCC_{1p}$, where p is the number of 2-input gates at level 1. We can represent the output of each gate in level 1 in terms of SCC . Thus, $p_{z_{11}} = f(SCC_{11}), p_{z_{12}} = f(SCC_{12}), p_{z_{13}} = f(SCC_{13}), \dots, p_{z_{1p}} = f(SCC_{1p})$ become functions of the corresponding SCC . These outputs are again inputs to certain gates in the next level of the circuit. The output from any gate in the intermediate level, is thus, in turn a function of SCC of primary input pairs.

Let the i th intermediate level consists of q interconnected gates. The output Z_{i1} from gate 1 in i th level is $p_{Z_{i1}} = f(p_{Z_{(i-1)k}}, p_{Z_{(i-1)j}})$ where

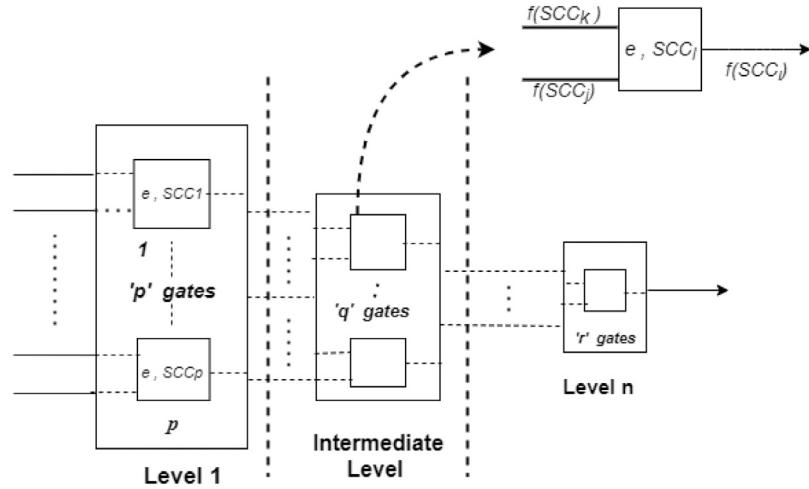


Fig. 12. Block Diagram representation of Multi-level MISO SLC.

$Z_{(i-1)k}$ and $Z_{(i-1)j}$ are the outputs from k th and j th gate at the $(i-1)$ th level. Thus, $p_{Z_{i1}} = f(p_{Z_{(i-2)l}}, p_{Z_{(i-2)m}}, p_{Z_{(i-2)n}}, p_{Z_{(i-2)r}})$, where $p_{Z_{(i-2)l}}$ and $p_{Z_{(i-2)m}}$ are the outputs of l th and m th gate of $(i-2)$ th level that are inputs to k th gate in $(i-1)$ th level and $p_{Z_{(i-2)q}}$ and $p_{Z_{(i-2)r}}$ are the outputs from q th and r th gate of $(i-2)$ th level that are connected to j th gate in $(i-1)$ th level. These outputs are again functions of SCCs of their corresponding inputs, such that $p_{Z_{i1}} = f(SCC_{(i-2)l}, SCC_{(i-2)m}, SCC_{(i-2)q}, SCC_{(i-2)r})$. Thus, tracing the interconnected sub-networks, $p_{Z_{i1}}$ is written as, $p_{Z_{i1}} = f(SCC_{11}, SCC_{12}, SCC_{13}, \dots, SCC_{1p})$ of traced inputs at the primary level.

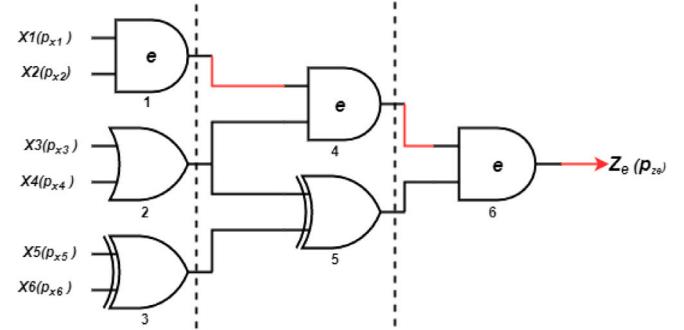
Let transient faults at a certain level contribute to a shift in the initial assumption of SCC at the inputs, that effect the succeeding levels leading to erroneous results. We take into account the intermediate correlation SCC_{i-1} present between $Z_{(i-1)k}$ and $Z_{(i-1)j}$ if the circuit is non-faulty. Let, $Z'_{(i-1)k}$ and $Z'_{(i-1)j}$ are the modified output values on account of errors from k th and j th faulty gates at level $(i-1)$. These result in a change in the number of 1's present in the bitstream. Let, probability of $Z_{(i-1)k}$ is changed from $n_{(i-1)k}$ to $n'_{(i-1)k}$ and $Z_{(i-1)j}$ is changed from $n_{(i-1)j}$ to $n'_{(i-1)j}$. Eventually, SCC_{i1} is modified to SCC'_{i1} which modifies the probability of Z_{i1} from $p_{Z_{(i-1)j}}$ to $p_{Z'_{(i-1)j}}$. Thus, $p_{Z'_{(i-1)j}} = (1 + SCC'_{(i-1)j})p_{Z'0_{(i-1)j}} - SCC'_{(i-1)j}p_{Z'(+1)_{(i-1)j}} = n'_{(i-1)j}/n$. Similarly, $p_{Z'_{(i-1)k}}$ can be written as $p_{Z'_{(i-1)k}} = (1 + SCC'_{(i-1)k})p_{Z'0_{(i-1)k}} - SCC'_{(i-1)k}p_{Z'(+1)_{(i-1)k}} = n'_{(i-1)k}/n$.

It is observed that the effect of transient error is reflected in an overall change in the probability of SNs. This suggests the dependence of MSE on SCC . Thus, a suitable technique can be investigated that adapts to the change in the assumption of correlation and tries to minimize the observed error at the output by using $ReCo$ method. It relocates SCC at several levels to counterbalance the change in initial assumption of SCC due to transient faults. We trace faults and identify faulty gates in the circuit and remodel $SCCs$ by modifying input vectors of the primary inputs of the sub network in level 1.

In noisy operating conditions, errors in output are assumed to be primarily contributed by one or more faulty gates in the circuit. For a multilevel MISO circuit shown in Fig. 13, the number of SLEs that undergo $ReCo$ correction primarily depends on the number of faulty gates and the probability of error. Thus, it is important to identify faulty paths in the circuit. We generalize the procedure for fault correction in *MISO* circuits as follows:

- Check if the MSE of the circuit is within the tolerable limit $\delta (<= 10^{-3})$ or not. If not,

i **Determine Faulty gates:** Generate T input test vectors n times and the output at each gate is observed. The error rate is estimated by the number of faulty outputs for n inputs using **FaultEvaluation(CIRCUIT)**.

Fig. 13. A sample MISO circuit showing gates $G1, G4, G6$ faulty.

ii **Determine input SLEs corresponding to faulty output node:**

Input gates that are connected to the faulty output node, denoted by **IsConnected()**, are selected and are stored in an array **FC_I_gate**.

iii **Register SLEs based on priority:** The gates based on their priority values from left (highest) to right (lowest); {*XOR, AND, OR*} are sorted using **PrioritySort()** and are stored in an array **S_FC_I_gate**.

iv **Selection of SLEs for ReCo:** Pop gate from the priority list and alter SCC using **ReCo()** (Algorithm 1). Calculate if, $MSE \leq \delta$, return corresponding SCC_i . If not, invoke next SLE from the sorted priority list.

From the previous discussion, it can be inferred that the XOR gate is most susceptible to changes in correlation and can reduce the overall MSE substantially compared to other correlation sensitive SLEs. So XOR gate line up the highest in the correlation-sensitivity index. If a single gate does not suffice then we proceed for combinations from the list **L_MSE[]**. Suppose, **PrioritySort()** list consists of {*XOR, AND1, AND2*} and *AND2* generates less MSE compared to *AND1*, then we combine *XOR* and *AND2* to find minimum MSE . This condition is often guided by the position of the gate(s) in the circuit. The whole analysis is carried out to improve the accuracy and reduce the number of correlators in the circuit.

It is observed that the MSE is proportional to the number of faulty gates and the error rate p_e . For any error observed at the output, the error can be due to transient error at the gate itself or due to the error being propagated from the previous stage or the both. We consider different cases of fault propagation in Fig. 13 where the path P is indicated in red colour.

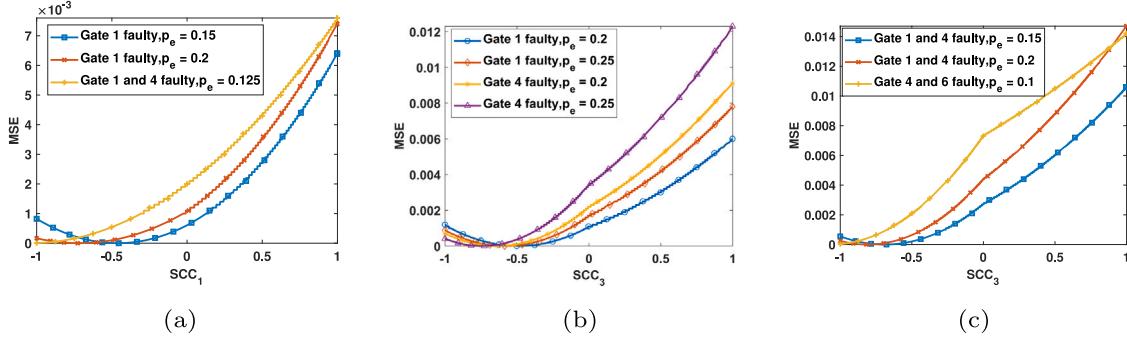


Fig. 14. Graphs showing different values of induced correlation to achieve minimum MSE for different gates faulty at different error rates in Fig. 13.

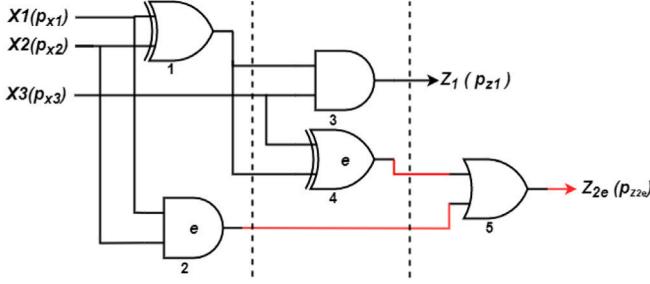


Fig. 15. A sample MIMO circuit showing gates $G2, G4$ faulty.

It is observed from Fig. 14(a),(b) that for a single faulty gate, $G1/G4$ with $p_e \leq 0.20$, single gate ReCo analysis, SCC_1 or SCC_3 will suffice to minimize MSE. For multiple gates faulty, as for example, $G1, G4$ faulty with $p_e \leq 0.125$ same logic can comply. Thus, in both cases, the error can be minimized without being thoroughly guided by the priority-rule of SLE (excluding OR). But $G1, G4$ faulty at $p_e = 0.2$ fault tolerance can be achieved using prioritized SLE (SCC_3) only as shown in Fig. 14(c) (red).

For a larger number of gates faulty even at low error rates, treatment of prioritized SLE is obligatory. Thus, $G1, G4, G6$ faulty at $p_e \leq 0.125$, MSE can be minimized using single gate ReCo (SCC_3). But the same with $p_e = 0.20$ we invoke dual gate ReCo (SCC_3, SCC_1) as shown in Fig. 18(d). When $G4$ and $G6$ are faulty with $p_e \leq 0.25$ we invoke dual ReCo (SCC_3, SCC_1) as shown in Fig. 18(a)–(c). It is observed that there is a finite error of 0.01 when $G1, G4, G6$ are faulty at a rate of 0.25 even after dual ReCo analysis of prioritized SLEs. Table 1 is given to comprehend the nature of the analysis and the results obtained in the proposed work.

It is observed that the possibility of error reduction is motivated by several factors; the nature and number of faulty gates as well as the arrangement of gates in the circuit. As AND gate is the most susceptible to soft errors, an AND gate in place of the XOR gate would contribute to larger MSE. These factors coupled with the probability of error play a pivotal role in determining the circuits' resilience towards soft errors. There is also a slight dependence on input probability values i.e. p_x and p_y as indicated in Fig. 7(a)–(c). The worst-case time complexity of Algorithm 2 is calculated as $O(G) + O(F) + O\{I\log(I)\} + O(I + I^2)$, where, G is the number of gates in the circuit and F is the number of faulty gates.

5.1.1. ReCo II: Error minimization for Multi-input and Multi-output (MIMO) SLC

We have argued that ReCo analysis of the prioritized SLEs can bring down MSE close to 0. We will see that in particular situations that can deviate from this initial assumption. When non-faulty gates converge to a different output node, ReCo analysis of primary SLEs may give

undesired results. The condition can be best described with the help of a Multiple-input–Multiple-Output circuit shown in Fig. 15. The circuit has two distinct outputs Z_1 and Z_2 with probabilities p_{z_1} and p_{z_2} . Consider two faulty gates $G2$ and $G4$ that converge to p_{z_2} i.e., the output of gate 5 and the output is modified to p_{z_2e} . We assume that there are no faulty gates in the path that converges to p_{z_1} . One way of suppressing the propagation of faulty results to the non-faulty output node is to perform ReCo analysis at the inputs of faulty gates only. This avoids analysis of primary SLEs that are explicitly connected to non-faulty outputs.

It is interesting to note that for $G2, G4$ faulty with an error rate ≤ 0.2 , the MSE at the output Z_2 can be reduced to 0 by doing ReCo analysis at either of the faulty gates, SCC_2 or SCC_4 , as shown in Fig. 16(a). We may perform ReCo at any faulty gate location, regardless of the precedence rule as shown in Fig. 16(b). For $p_e > 0.2$, however, MSE can only be reduced to 0 using Algorithm 3. Modelling correlation between inputs of gate $G4$ (which is the priority in this case), can reduce MSE to 0 with a soft error margin of 0.3. At $p_e = 0.325$, we invoke dual ReCo (SCC_4, SCC_2) to observe the error free behaviour as shown in Fig. 18(e). ReCo is performed specifically at the faulty gate site to prevent the generation of erroneous output at a node that is preceded by non-erroneous output nodes. The modelling of SCCs at the primary gates $G1$ is avoided without affecting the output logic level at Z_1 . This is contrary to the selection criteria in Algorithm 2. It attempts to minimize the number of correlator circuits for larger error rates 0.30 ($l = 1$). It is observed that MSE can be reduced to zero even for very high error rates, i.e., $p_e = 0.325$ using dual ReCo analysis. The results of analysis are recorded in Table 2. The worst-case time complexity of Algorithm 3 is calculated as $O(G) + O\{F\log(F)\} + O(F + F^2)$, where, G is the number of gates in the circuit and F is the number of faulty gates.

6. Experimental results and discussion

We can now identify the key factors on which the whole analysis is hinged upon. The magnitude and polarity of induced correlation, as well as the number (l) of ReCo blocks, depend on several underlying factors. The predominant factor is the amount of transient error in the circuit. As transient error increases, the MSE increases exponentially (see graphs in Fig. 1(a)). With higher MSE the value of l tends to be larger. For $G1, G4, G6$ faulty at $p_e \leq 0.125$ error can be reduced to 0 using a single ReCo block (SCC_3). The $p_e \geq 0.20$ error, however, can be reduced to 0 with $l = 2$ as shown in Table 1. This is because MSE is higher in the second case.

It is identified that as the number of faulty gates in the circuit increases, the value of l increases to account for the increased MSE. When $G1$ faulty and $G1, G4, G6$ faulty with the same error rate, i.e., 0.20, the error-free output is obtained respectively with $l = 1$ (SCC_3/SCC_1) and $l = 2$ (SCC_3, SCC_1). In addition, the proliferation of the SLE with the highest priority in the input panel increases the possibility of reducing the error to zero. Therefore, if AND gate $G1$ is replaced with

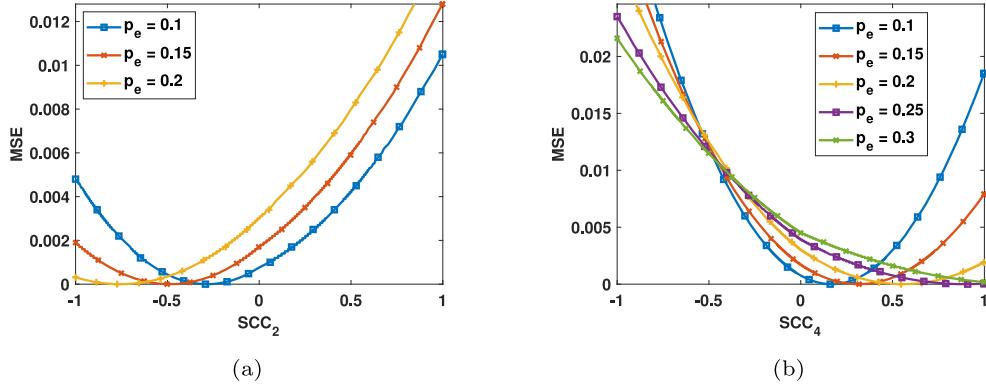


Fig. 16. Graphs showing different values of induced correlation to achieve minimum MSE for $G2, G4$ faulty at different error rates in Fig. 15.

Table 1

Comparison of MSE with and without ReCo of the circuit in Fig. 13. MSE_{min} represents minimum MSE .

	Faulty gate(s)	MSE without ReCo	with ReCo (Single/Dual gate)				
			SCC_1	SCC_3	SCC_1, SCC_3	SCC_2, SCC_3	
Error = 0.125	Gate 1	0.00042	SCC_i MSE_{min}	-0.36 0	-0.61 0	-0.19, -0.16 0	-0.26, 0.33 0
	Gate 4	0.00085	SCC_i MSE_{min}	-0.51 0	-0.45 0	-0.27, -0.23 0	-0.34, -0.38 0
	Gate 1, 4	0.002	SCC_i MSE_{min}	-1 0	-0.61 0	-0.4471, -0.4001 0	-0.51, -0.82 0
	Gate 1, 4, 6	0.016	SCC_i MSE_{min}	-1 0.0095	-1 0	-0.9, -0.9 0	-1, 0 0.001
Error = 0.20	Gate 1	0.0011	SCC_i MSE_{min}	-0.717 0	-0.4896 0	-0.36, -0.27 0	-0.5, -0.49 0
	Gate 4	0.0022	SCC_i MSE_{min}	-1 0	-0.6385 0	-0.45, -0.41 0	-1, -0.1 0
	Gate 1, 4	0.0044	SCC_i MSE_{min}	-1 0.0015	-0.8078 0	-0.546, -0.7 0	-0.524, -0.68 0
	Gate 1, 4, 6	0.0375	SCC_i MSE_{min}	-1 0.0314	-1 0.0108	-1, -1 0	-1, -0.15 0.01
Error = 0.25	Gate 1	0.0017	SCC_i MSE_{min}	-1 0.0114	-0.5813 0	-0.44, -0.38 0	-0.47, -0.83 0
	Gate 4	0.0034	SCC_i MSE_{min}	-1 0.0004	-0.743 0	-52, -0.55 0	-1, -0.1 0
	Gate 1, 4	0.049	SCC_i MSE_{min}	-1 0.036	-0.8987 0	-1, -1 0	-1, -0.09 0.0045
	Gate 1, 4, 6	0.1317	SCC_i MSE_{min}	-1 0.041	-1 0.023	-1, -1 0.01	-1, 0 0.02

Table 2

Comparison of MSE with and without ReCo of the circuit in Fig. 15. MSE_{min} represents minimum MSE .

	Faulty gates	MSE without ReCo	with ReCo (single or dual gate(s))			
			SCC_2	SCC_4	SCC_2, SCC_4	
Error = 0.25	Gates 2,4	0.003	SCC_i MSE_{min}	-0.9 0.001	-0.33 0	-0.2, -0.4 0
Error= 0.30	Gates 2,4	0.004	SCC_i MSE_{min}	-0.93 0.002	-0.87 0	-0.15, -1 0

an XOR gate, MSE can be reduced to 0, even if, $G1, G4, G6$ are faulty at $p_e = 0.25$.

The observed MSE is greatly dependent on the nature of faulty gates. If AND gate $G4$ is replaced in the circuit with an XOR gate, the overall MSE is reduced from 0.0375 to 0.02 when $G1, G4, G6$ are faulty at $p_e = 0.20$. This is because XOR is least susceptible to transient errors. The location of faulty gates within the circuit has also an impact on the overall MSE . A faulty gate distant from the periphery of the primary input and closer to the output contributes to a larger MSE . As for example, $G1$ faulty at an error rate $p_e = 0.2$ gives MSE as 0.0011, whereas $G6$ faulty at the same error rate gives larger MSE (0.0237).

This is because $G6$ is distant from the primary input side than $G1$. It is also implicit from Fig. 1(c), (d) and Fig. 2(c), (d) that the initial assumption of SCC also plays a significant role in determining the exact operating point of SCC for a given circuit. In the current experimental setup as in Fig. 15 the error rates up to 0.3 can be handled accurately and this is determined by the number of faulty gates in the circuit (≤ 2) and location of faulty gates (closer to the input side) as shown in Fig. 18(e). A block diagram illustrating the interdependence of these parameters is shown in Fig. 17.

For complex circuits with multiple faulty gates, we have introduced a priority-based selection scheme of SLEs showing promising results.

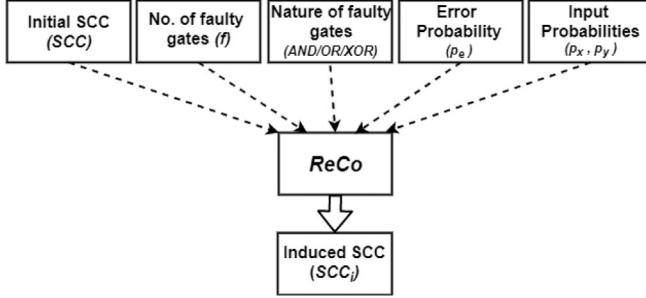


Fig. 17. Factors influencing the polarity and the magnitude of the induced SCC.

In our investigation, the XOR gate is a prime component because of its high correlation sensitivity. Assuming a priority-based strategy, the bare minimum number of *ReCo* blocks needed to achieve a minimum *MSE* is found as $l = 1$. The only exception is $G1, G4, G6$ faulty at $p_e = 0.25$, which requires two *ReCo* blocks to achieve minimum *MSE*. The graphs of Fig. 19 are obtained with $G1, G4, G6$ faulty at different error rates. The deviation in output from the actual value (without error) using the proposed scheme is much less compared to the non-priority-based approach. It has been found that this method can deal with high error rates while using less hardware. Furthermore, the desired value can be achieved with fewer iterations when using a priority-based approach. The deviation graph with the priority-based (red) approach is obtained with one *ReCo* block whereas the deviation with non-priority based approach (gray) is obtained with two *ReCo* blocks to model the output. The blue line represents to the observed error (without *ReCo*). This demonstrates the efficacy of the proposed priority-based approach in terms of hardware design. The efficacy of the proposed method relies heavily on the presence of correlation sensitive logic blocks in the circuit. If there are no correlation sensitive logic blocks in the circuit the method fails drastically. Also, care needs must be taken while placing the *ReCo* blocks at specific targets without which errors could be propagated to non-erroneous output node.

As there is no standard benchmark stochastic circuit available in literature, we have implemented a complex matrix multiplication circuit using stochastic logic blocks under the assumption that the erroneous behaviour of the circuit is due to the cumulative error caused by the 12 constants and some of the gates being noisy due to the presence of transient error. A comparison of *MSE* using CEASE method in the presence of only constant error [15] and the proposed method in presence of both the constant error and transient error is shown in Table 4. Due to presence of both the errors in the circuit the original *MSE* was bit higher in our case, but after correction using *ReCo* method the *MSE* can be lowered significantly, which shows that the proposed method can potentially combat the effect of the duo.

In Table 3 estimates the number of *ReCo* blocks needed to minimize the *MSE* at a given error rate to indicate the efficiency of the proposed method in making a trade-off between error metrics and overheads. The cells marked in grey show the variations in *MSE* at a particular error rate. In cases, where it is not possible to achieve the lowest *MSE* using a single block, a second block is inserted to observe the error-free output. Increase in the number of *ReCo* blocks in the circuit increases hardware overhead, which can be minimized by using a priority-based approach. It is observed that even for very high degree of error rates with multiple gates faulty, it is possible to bring down the *MSE* within a limit using only two *ReCo* blocks in the circuit.

7. Case study: Contrast enhancement in images

In order to demonstrate the practicality and efficacy, we have implemented the proposed technique for contrast enhancement [31] to images of a standard publicly available contrast enhancement dataset,

Algorithm 2: *ReCo* analysis for MISO Circuits

```

1: Input: CIRCUIT // Circuit with n number of inputs.
   Output: MSE, SCCi
2: Variable Initialization:
   p_arr[ ] = {XOR, OR, AND} //Priority Sequence
   F_gate[ ] = {0} // number of faulty gates
   I_gate[ ] = {number of input gates}
   FC_I_gate[ ] = {0} // Input gates ∈ F_gate[ ]
   S_FC_I_gate[ ] = {0} // Sorted FC_I_gate[ ]
3: F_gate=FaultEvaluation(CIRCUIT)
   // Identify faulty gates in the circuit
4: FC_I_gate=IsConnected(I_gate, F_gate)
5: S_FC_I_gate=PrioritySort(FC_I_gate, p_arr)
6: for j = 1 to j <= maxElement (S_FC_I_gate) do
   [L_MSE[j], SCCi[j]] = ReCo(S_FC_I_gate[j])
   if (L_MSE[j] <= δ) then
   | return L_MSE[j], SCCi[j]
   end if
   end for
7: [S_Lgate, L_MSE] = Sort(L_MSE) //Sort MSE value along with their
   gate number
8: for j = 1 to j <= maxElement(S_FC_I_gate) do
   [MSE[j], SCCi[j]] = NewReco(S_Lgate, j + 1)
   if (MSE[j] <= δ) then
   | return MSE[j], SCCi[j]
   end if
   end for
9: return argminMSE[j] {MSE[j] , SCCi[j]}

```

Algorithm 3: *ReCo* analysis for MIMO Circuits

```

Input: CIRCUIT // Circuit with n number of inputs
Output: MSE, SCCi
Variable Initialization:
p_arr[ ] = {XOR, OR, AND} //Priority Sequence
F_gate[ ] = {0} // number of faulty gates
S_F_gate[ ] = {0} // Sorted F_gate[ ]
F_gate=FaultEvaluation(CIRCUIT)
// Identify faulty gates in the circuit
S_F_gate=PrioritySort(F_gate, p_arr)
for j = 1 to j <= maxElement(S_F_gate) do
   [L_MSE[j], SCCi[j]] = ReCo(S_F_gate[j])
   if (L_MSE[j] <= δ) then
   | return L_MSE[j], SCCi[j]
   end if
   end for
[S_Lgate, L_MSE] = Sort(L_MSE) //Sort MSE value along with their gate
number
for j = 1 to j <= maxElement(S_F_gate) do
   [MSE[j], SCCi[j]] = NewReco(S_Lgate, j + 1)
   if (MSE[j] <= δ) then
   | return MSE[j], SCCi[j]
   end if
   end for
return argminMSE[j] {MSE[j] , SCCi[j]}

```

CEED2016 [30] which contains 30 benchmark images. The output contrast enhanced images after introducing error and its subsequent revision using the proposed methodology are shown in Fig. 20 using two randomly selected images from the CEED2016 dataset. Different evaluation metrics such as Multi Scale Structural Similarity Index (MS-SSIM) [32], Entropy [33], Peak signal-to-noise ratio (PSNR) [34] and Contrast-to-Noise Ratio (CNR) [34] are calculated on the entire CEED2016 dataset to measure the similarity between the enhanced images and the ground truth image using our methods. The results are shown in Fig. 21. It is observed that all the metrics of the enhanced images using the contrast stretching technique are improved after the correction of errors. A priority-based strategy is utilized for this

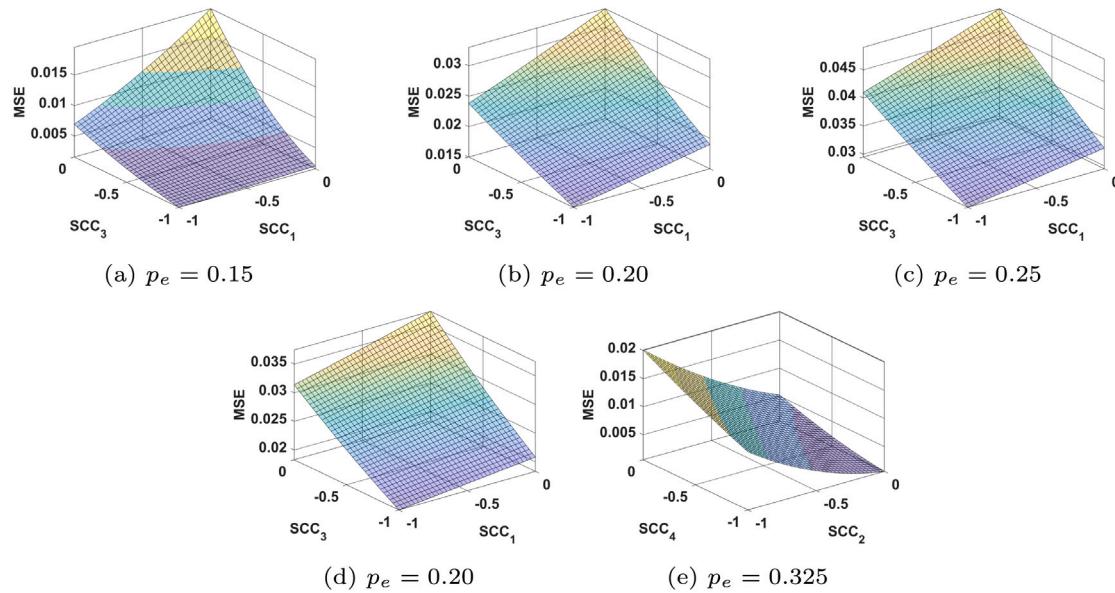


Fig. 18. Error minimization using dual ReCo analysis at different error rates when; (a), (b), (c) $G4, G6$ faulty and (d) $G1, G4, G6$ faulty in Fig. 13; (e) $G2, G4$ faulty in Fig. 15.

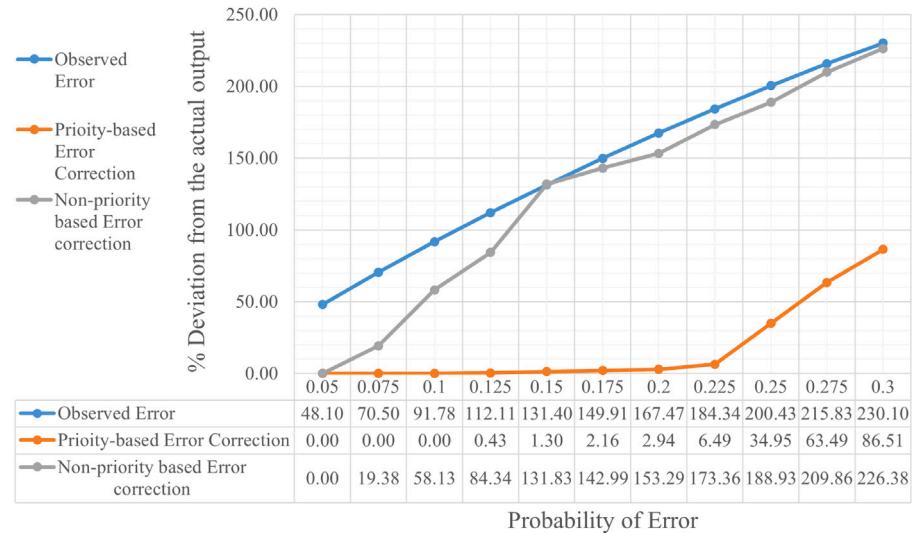


Fig. 19. Deviation in output of two correction approaches from the actual value (without error) of the circuit in Fig. 13.

Table 3
Minimization of MSE using varying number $ReCo$ blocks at different error rates.

SL#	Fault Rate Fault Location	Minimization of MSE using varying number $ReCo$ blocks at different error rates.					
		$p_e = 0.1$		$p_e = 0.2$		$p_e = 0.3$	
		#ReCo blocks(l)	Min. MSE	# ReCo blocks(l)	Min. MSE	# ReCo blocks(l)	Min. MSE
1	G1	l=1	0	l=1	0	l=1	0
		l=2	0.007	l=2	0	l=2	0
2	G1,G4	l=1	0	l=1	0	l=1	0.007
		l=2	0	l=2	0	l=2	0
3	G1,G4,G6	l=1	0	l=1	0.01	l=1	0.05
		l=2	0	l=2	0	l=2	0.027

purpose, where two SLEs are chosen to achieve an error-tolerant circuit behaviour. From Fig. 21(a), it is observed that the MS-SSIM indices using the proposed ReCo method are considerably higher compared to the enhanced image with the error, indicating that the proposed

methodology provides faithful results even in transient error scenarios. The entropy of the different images after image enhancement are shown in Fig. 21(b). The figure shows the entropy of the images (after correction) are close to the ground truth images compared to the images

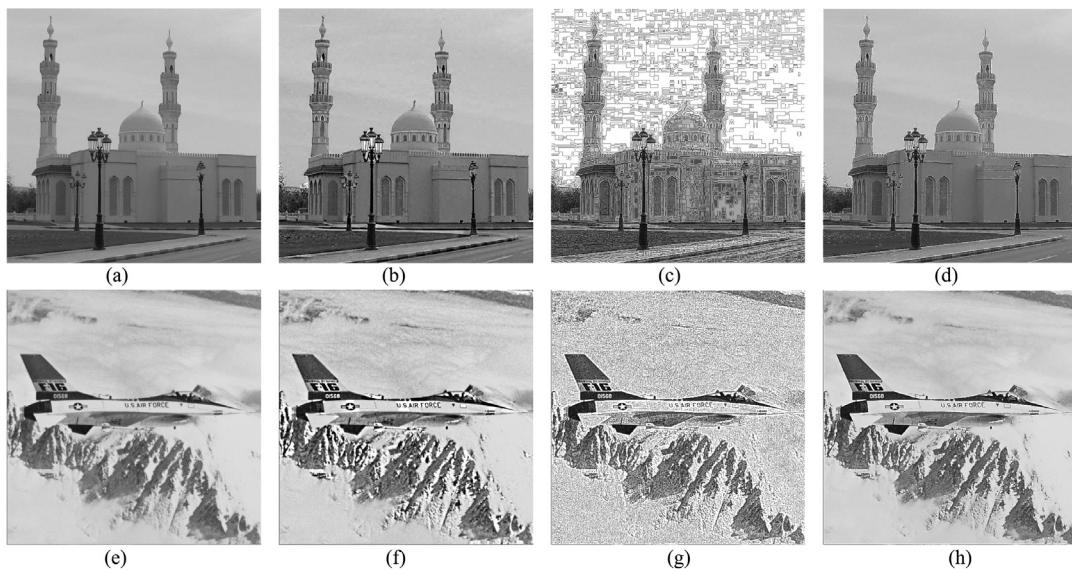


Fig. 20. Samples images of CEED2016 dataset [30] before and after Contrast Stretching operations along with ground truth. (a,e) Original images, (b,f) Ground truth images, (c,g) Enhanced images with error, and (d,h) Corrected images using the proposed framework.

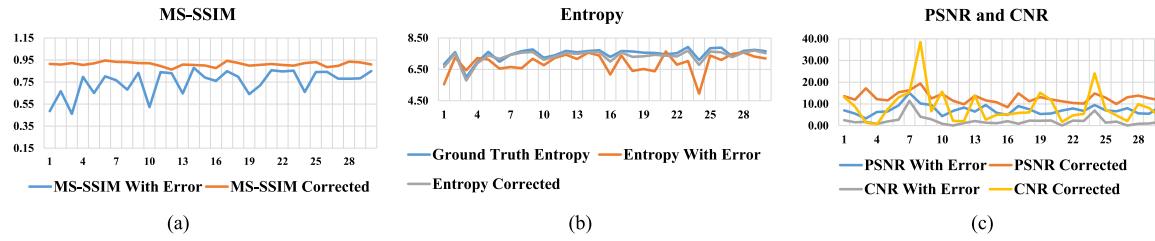


Fig. 21. Comparisons of different metrics calculated after Contrast Stretching operations on the images of CEED2016 dataset [30] with error and after the error correction using the proposed methodology.

Table 4
Performance comparison on Complex Matrix Multiplication using CEASE [15] and the proposed method.

Mean squared error	Matrix Multiplication	
	CEASE (Constant error)	ReCo (Constant error + Transient error)
Original	8×10^{-3}	10.75×10^{-3}
After correction	2×10^{-3}	0.44×10^{-3}

obtained with error. The PSNR and CNR values of the images are also shown in Fig. 21(c). The proposed method thus can be implemented with lower hardware cost in various image processing applications.

8. Conclusion

Recent applications of stochastic computing involve noisy operating conditions leading to incorrect results at times. The source of inaccuracy has been predominantly traced to transient errors. In this work, we have progressively varied the transient error probabilities for single gates and observed their effect on the *MSE* of these gates. Attempts are made to formulate the process within a mathematical framework. The study on the effect of varying correlation on the *MSE* was extended to realistic multi-level circuits where single or multiple gates are subjected to transient errors. For such circuits, we have modified the framework to minimize the overall *MSE*. Algorithm 2 introduces a priority-based approach of choosing *SLE* to reduce the number of correlator circuits and to obtain the desired level of accuracy quickly under noisy operating conditions. Inevitably, there are conflicts in constraints in different applications, which are handled elegantly.

Algorithm 3 eliminates this issue for a MIMO circuit by introducing correction blocks at fault specified nodes only. Both these algorithms have been observed to handle noise and yield accurate results, even at high transient error rates. In our future work, we will explore other variants of these algorithms to achieve even better accuracy using lesser number of *ReCo* blocks. Also, equivalent circuits can be developed with a given set of conditions to ease the task of error minimization at a lesser hardware cost.

CRediT authorship contribution statement

Shyamali Mitra: Conceptualized the whole idea, Implemented the algorithms in the circuit level, Mathematical analysis of the proposed method. **Sayantan Banerjee:** Comparison with other algorithms, Finding algorithm complexity, Generated relevant graphs. **Mrinal Kanti Naskar:** Reviewed the paper, Providing suggestions.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

- [1] Gaines BR. Stochastic computing. In: Proceedings of the april 18-20, 1967, spring joint computer conference. 1967, p. 149–56.
- [2] Alaghi A, Hayes JP. Survey of stochastic computing. ACM Trans Embed Comput Syst (TECS) 2013;12(2s):1–19.
- [3] Najafi MH, Salehi ME. A fast fault-tolerant architecture for sauvola local image thresholding algorithm using stochastic computing. IEEE Trans Very Large Scale Integr (VLSI) Syst 2015;24(2):808–12.
- [4] Li P, Lila DJ. Using stochastic computing to implement digital image processing algorithms. In: 2011 IEEE 29th international conference on computer design. IEEE; 2011, p. 154–61.
- [5] Alaghi A, Qian W, Hayes JP. The promise and challenge of stochastic computing. IEEE Trans Comput-Aided Des Integr Circuits Syst 2017;37(8):1515–31.
- [6] Nicolaidis M. Design for soft error mitigation. IEEE Trans Device Mater Reliab 2005;5(3):405–18.
- [7] Xiao R, Chen C. Gate-level circuit reliability analysis: A survey. VLSI Des 2014;2014. <http://dx.doi.org/10.1155/2014/529392>.
- [8] Vallero A, Savino A, Chatzidimitriou A, Kaliorakis M, Kooli M, Riera M, et al. SyRA: Early system reliability analysis for cross-layer soft errors resilience in memory arrays of microprocessor systems. IEEE Trans Comput 2019;68(5):765–83. <http://dx.doi.org/10.1109/TC.2018.2887225>.
- [9] Xu X, Ban T, Li Y. SPLM: A flexible and accurate reliability assessment model for logic circuits. J Circuits Syst Comput 2019;28(02):1950032. <http://dx.doi.org/10.1142/S0218126619500324>, arXiv:10.1142/S0218126619500324.
- [10] Miskov-Zivanov N, Marculescu D. Multiple transient faults in combinational and sequential circuits: A systematic approach. IEEE Trans Comput-Aided Des Integr Circuits Syst 2010;29(10):1614–27.
- [11] Miskov-Zivanov N, Marculescu D. A systematic approach to modeling and analysis of transient faults in logic circuits. In: 2009 10th international symposium on quality electronic design. IEEE; 2009, p. 408–13.
- [12] Miskov-Zivanov N, Marculescu D. MARS-C: modeling and reduction of soft errors in combinational circuits. In: Proceedings of the 43rd annual design automation conference. 2006, p. 767–72.
- [13] Rajaraman R, Kim J, Vijaykrishnan N, Xie Y, Irwin MJ. SEAT-LA: A soft error analysis tool for combinational logic. In: 19th international conference on VLSI design held jointly with 5th international conference on embedded systems design. IEEE; 2006, p. 1–4.
- [14] Cai S, He B, Wang W, Liu P, Yu F, Yin L, et al. Soft error reliability evaluation of nanoscale logic circuits in the presence of multiple transient faults. J Electron Test 2020;36(4):469–83.
- [15] Ting P, Hayes JP. Eliminating a hidden error source in stochastic circuits. In: 2017 IEEE international symposium on defect and fault tolerance in VLSI and nanotechnology systems. 2017, p. 1–6. <http://dx.doi.org/10.1109/DFT.2017.8244436>.
- [16] Krishnaswamy S, Viamontes GF, Markov IL, Hayes JP. Probabilistic transfer matrices in symbolic reliability analysis of logic circuits. ACM Trans Des Autom Electron Syst (TODAES) 2008;13(1):1–35.
- [17] Alaghi A, Ting P, Lee VT, Hayes JP. Accuracy and correlation in stochastic computing. In: Stochastic computing: techniques and applications. Springer; 2019, p. 77–102.
- [18] Krishnaswamy S, Viamontes GF, Markov IL, Hayes JP. Accurate reliability evaluation and enhancement via probabilistic transfer matrices. In: Design, automation and test in europe. IEEE; 2005, p. 282–7.
- [19] Anderson JH, Hara-Azumi Y, Yamashita S. Effect of LFSR seeding, scrambling and feedback polynomial on stochastic computing accuracy. In: 2016 design, automation & test in europe conference & exhibition. IEEE; 2016, p. 1550–5.
- [20] Alaghi A, P. Hayes J. Exploiting correlation in stochastic circuit design. In: 2013 IEEE 31st international conference on computer design. IEEE; 2013, p. 39–46.
- [21] Ahlgren P, Jarneving B, Rousseau R. Requirements for a cocitation similarity measure, with special reference to Pearson's correlation coefficient. J Am Soc Inf Sci Technol 2003;54(6):550–60.
- [22] Budhwani RK, Ragavan R, Senteys O. Taking advantage of correlation in stochastic computing. In: 2017 IEEE international symposium on circuits and systems. IEEE; 2017, p. 1–4.
- [23] Alaghi A, Hayes JP. On the functions realized by stochastic computing circuits. In: Proceedings of the 25th edition on great lakes symposium on VLSI. 2015, p. 331–6.
- [24] Kanoh T. Absolute value calculating circuit having a single adder. Google Patents; 1990, US Patent 4, 953, 115.
- [25] Lee VT, Alaghi A, Ceze L. Correlation manipulating circuits for stochastic computing. In: 2018 design, automation & test in europe conference & exhibition. IEEE; 2018, p. 1417–22.
- [26] Shrivakumar P, Kistler M, Keckler SW, Burger D, Alvisi L. Modeling the effect of technology trends on the soft error rate of combinational logic. In: Proceedings international conference on dependable systems and networks. IEEE; 2002, p. 389–98.
- [27] Alaghi A, Chan W-TJ, Hayes JP, Kahng AB, Li J. Trading accuracy for energy in stochastic circuit design. ACM J Emerg Technol Comput Syst (JETC) 2017;13(3):1–30.
- [28] Chen T-H, Alaghi A, Hayes JP. Behavior of stochastic circuits under severe error conditions. It-Inform Technol 2014;56(4):182–91.
- [29] Chen T-H, Hayes JP. Analyzing and controlling accuracy in stochastic circuits. In: 2014 IEEE 32nd international conference on computer design. IEEE; 2014, p. 367–73.
- [30] Qureshi MA, Sdiri B, Deriche M, Alaya-Cheikh F, Beghdadi A. Contrast enhancement evaluation database (CEED2016), vol. 3. 2017, <http://dx.doi.org/10.17632/3hfz6vvkm.3>, URL <https://data.mendeley.com/datasets/3hfz6vvkm>.
- [31] Dhawan AP, Buelloni G, Gordon R. Enhancement of mammographic features by optimal adaptive neighborhood image processing. IEEE Trans Med Imaging 1986;5(1):8–15. <http://dx.doi.org/10.1109/tmi.1986.4307733>.
- [32] Wang Z, Simoncelli EP, Bovik AC. Multiscale structural similarity for image quality assessment. In: The thirty-seventh asilomar conference on signals, systems & computers, 2003, Vol. 2. 2003, p. 1398–402.
- [33] Gonzalez RC, Woods R, Eddins SL. Digital image processing using MATLAB (R). Upper Saddle River, NJ: Pearson; 2003.
- [34] Welvaert M, Rosseel Y. On the definition of signal-to-noise ratio and contrast-to-noise ratio for fMRI data. In: Yacoub E, editor. PLoS ONE 2013;8(11):e77089. <http://dx.doi.org/10.1371/journal.pone.0077089>.