Logical stochastic resonance

Adi R. Bulsara a,*, Anna Darib, William L. Ditto b, K. Muralic, Sudeshna Sinhed

a SPAWAR Systems Center Pacific, San Diego, CA 92152-5001, USA
b Ira A. Fulton School of Engineering, Arizona State University, Tempe, AZ 85287-9309, USA
c Department of Physics, Anna University, Chennai 600 025, India
d Institute of Mathematical Sciences, Taramani, Chennai 600 113, India
e Indian Institute of Science Education and Research, Mohali, Transit Campus: MGSIPAP Complex, Sector 26 Chandigarh, India

ABSTRACT

In a recent publication it was shown that, when one drives a two-state system with two square waves as input, the response of the system mirrors a logical output (NOR/OR). The probability of obtaining the correct logic response is controlled by the interplay between the noise-floor and the nonlinearity. As one increases the noise intensity, the probability of the output reflecting a NOR/OR operation increases to unity and then decreases. Varying the nonlinearity (or the thresholds) of the system allows one to morph the output into another logic operation (NAND/AND) whose probability displays analogous behavior. Thus, the outcome of the interplay of nonlinearity and noise is a flexible logic gate with enhanced performance. Here we review this concept of “Logical Stochastic Resonance” (LSR) and provide details of an electronic circuit system demonstrating LSR. Our proof-of-principle experiment involves a particularly simple realization of a two-state system realized by two adjustable thresholds. We also review CMOS implementations of a simple LSR circuit, and the concatenation of these LSR modules to emulate combinational logic, such as data flip-flop and full adder operations.

1. Introduction

Considerable research activity has centered around understanding how noise and nonlinearity cooperate in a dynamical system to produce novel effects. Such studies help us understand how complex systems behave and evolve. Also, importantly, the successful manipulation and control of the interplay between noise and nonlinearity allows us to obtain enhanced performance in different contexts. Stochastic resonance (SR) provides one such example [1,2], which yields an enhanced (suitably quantified) response to weak periodic input signals.

The focus of this paper is a remarkable, related, phenomenon we have recently observed: we found that the response of a simple threshold detector to input signals, consisting of two random square waves, in an optimal band of noise, is a logical combination of the two input signals. This phenomenon has been dubbed Logical Stochastic Resonance (LSR) [3]. The motivation for further studying LSR stems from an issue that is receiving considerable attention today: as computational devices and platforms continue to shrink in size, one necessarily encounters fundamental circuit noise characteristics. Since this (performance degrading) noise cannot be suppressed or eliminated, an understanding of the interplay between a device noise-floor and its nonlinearity must play an increasingly crucial role in the design of computational devices.

In this article we first review the ideas propounded in [3]. We also include some new results on LSR under multiplicative noise, in order to demonstrate the generality of the idea over a range of systems and operating conditions. Further, we provide extensive details of the experimental implementation of LSR, as well as results from the concatenation of LSR modules to emulate combinational logic, e.g. data flip-flop and full adder operations.

The plan of the paper is as follows: Section 2 discusses the general concept, and Section 3 recalls the explicit demonstration of LSR with simple 1D systems, including a brief presentation of some preliminary results on systems containing multiplicative noise. In Section 4 we discuss the implementation of assorted combinational and sequential logic operations, and conclude with a presentation of experimental details in Section 5. The treatment in this article is purely numeric and experimental; the idea is to show that the phenomenon exists, is robust and, actually, can assist in reliable logic operations in noisy nonlinear gate circuits. A complete theoretical description is underway and will be the subject of forthcoming publications.

2. LSR: The general principle

Consider a general nonlinear dynamic system, given by:
where $x$ is a generic nonlinear function obtained via the negative gradient of a potential with two distinct stable energy wells. $I$ is a low amplitude input signal and $\eta(t)$ is an additive zero-mean Gaussian noise with unit variance, $D$ being the noise strength; the noise is taken to have correlation time smaller than any other time scale in the system, so that it may be represented, theoretically, as delta correlated. In practice, of course, the noise would be exponentially correlated, with the autocorrelation approaching the delta function in the limit of vanishing correlation time approaches zero.

A logical input–output correspondence can be realized by encoding $N$ inputs in $N$ square waves. Specifically, for two logic inputs, we drive the system with a low amplitude signal $I$, taken to be the sum of two trains of aperiodic pulses: $I_1 + I_2$, with $I_1$ and $I_2$ encoding the two logic inputs. The logic inputs can be either 0 or 1, giving rise to four distinct logic input sets $(I_1, I_2)$: $(0, 0)$, $(0, 1)$, $(1, 0)$ and $(1, 1)$. Now the input sets $(0, 1)$ and $(1, 0)$ give rise to the same $I$, thus the four distinct input conditions $(I_1, I_2)$ reduce to three distinct values of $I$. Hence, the input signal $I$, generated by adding two independent input signals, is a 3-level aperiodic waveform.

The output of the system is determined by its state e.g., the output can be considered a logical 1 if it is in one state (or potential energy well), and logical 0 if it is in the other. Specifically, the output corresponding to this 2-input set $(I_1, I_2)$, for a system with potential wells at $x_0 > 0$ and $x_0 < 0$, is taken to be 1 (or 0) when the system is in the well at $x_0$, and 0 (or 1) when the system is in the other well (at $x_0$). So when the state of the system switches, the output toggles from 0 to 1 and vice versa (see Figs. 1 and 2).

Here we will explicitly show that one observes, for a given set of inputs $(I_1, I_2)$, a logical output from this nonlinear system, in accordance with the truth tables of the basic logic operations shown in Table 1. A crucial observation is that this occurs consistently and robustly only in an optimal window of noise. For very small or very large noise the system does not yield any consistent logical output, in line with the basic tenets of SR. But in a reasonably wide band of moderate noise intensity, the system produces the desired logical outputs reliably.

### 3. An explicit example

We now explicitly demonstrate LSR with a representational nonlinear system, which we implement with a simple and efficient electronic analog circuit (described later in this paper):

$$\dot{x} = F(x) + I + D\eta(t) \tag{1}$$

with the nonlinear function $g(x)$ given by:

$$g(x) = \begin{cases} x_1^u & x < x_1^l \\ x_1 & x_1^l \leq x \leq x_0 \\ x_0 & x > x_0 \\ \end{cases} \tag{3}$$

where $x_1^l$ and $x_1^u$ are the upper and lower thresholds respectively and $\varepsilon$ is an asymmetrizing DC input (see Fig. 2a). The potential energy function is, of course, bistable and can be described by specific functional forms, depending on the dynamics of the system; we use the piecewise linear representation of (3) simply because it is easily and efficiently implemented in an analog circuit. The effective potential generated by the thresholding (see Fig. 2) is bistable with stable energy states (nominally potential minima), in the absence of the signal $I$, for $0 < x < \beta$, at:

$$x_\ast = \left(\beta x_1^u + \varepsilon\right)/\alpha$$

and

$$x_* = \left(\beta x_1^l + \varepsilon\right)/\alpha$$

This can be seen easily by finding the possible extremum solutions $(P(x) = 0)$ and examining $-dP/dx$ for these solutions, in the three different regions: (I) $x < x_1^l$, (II) $x_1^l < x < x_0$ and (III) $x > x_0$. In region 1, $P(x) = 0$ implies $-ax_1 + b = 0$, which gives solution

$$x_1 = \left(\beta x_1^u + \varepsilon\right)/\alpha$$

This is a minimum, as $-dP/dx = a > 0$. Similarly, in region III, $P(x) = 0$ implies $-ax_0 + b = 0$, which gives solution

$$x_0 = \left(\beta x_1^l + \varepsilon\right)/\alpha$$

Again, this is a minimum, as $-dP/dx = a < 0$. Lastly, in region II, $P(x) = 0$ implies $-ax_1 + b = 0$, which gives solution

$$x_{out} = \varepsilon/(\alpha - \beta)$$

This is a maximum, as $-dP/dx = a - \beta < 0$. So we have two potential wells at $x_\ast$ and $x_*$, separated by a barrier at $x_{out}$.

In this work, instead of manipulating the nonlinearity by varying $\alpha$ and $\beta$, we will vary the nonlinearity by simply changing the threshold $x_1^l$ and $x_1^u$ in Eq. (2). This allows the heights and asymmetry of the wells to be manipulated very efficiently in the circuit realization of the system.

We now quantify the input signal. With no loss of generality, consider the two inputs $I_1$ and $I_2$ to take value $-0.4$ when the logic input is 0, and value 0.4 when logic input is 1, namely the input signal $I = I_1 + I_2$ is a three-level square wave form: $-0.8$ corresponding to input set $(0, 0)$, 0 corresponding to input sets $(0, 1)$, $(1, 0)$ and 0.8 corresponding to input set $(1, 1)$. These square pulses are presented to the system in random sequence, with $I_1$ and $I_2$ switching levels in an uncorrelated aperiodic manner. This aperiodic stream of input pulses closely mimics input streams to logical elements in computing devices.

Fig. 3 shows the response of the system (2) for low, moderate and high levels of noise. We observe that, under moderate noise (panel 5 of Fig. 3), interpreting the state $x < 0$ as logic output 0 and the state $x > 0$ as logic output 1 clearly yields a stable logical AND whereas interpreting the state $x > 0$ as logic output 0 and the state $x < 0$ as logic output 1 yields a clearly defined reliable logical NAND. In a completely analogous way, by setting different thresholds we can realize clearly defined OR and NOR gates in almost the same optimal noise intensity regime as the previous case. Note that NAND and NOR are fundamental gates which can, in combination, yield all possible logical responses.
the correct output. This measure of the success of the logic operation, and when inputs \((0,1)/(1,0)\) are the distinguishing pairs, the system yields logic operations with near certain reliability, i.e., \(P(\text{logic}) \sim 1\). Also notice that, in contrast to the usual peak associated with SR [1], here we have a flat maximum in our performance measure, where the system yields almost 100% accuracy. So the optimal gate performance is robust, as it occurs over a wide range of noise.

Hence, under moderate noise one obtains a large, robust, asymmetric response to input signals: different distinct levels of input pulses can be mapped to a binary 0/1 output, determined by the system being in either one of the two widely separated wells. For instance, for parameter settings yielding NAND logic, the system driven by subthreshold input signals are realized only in the presence of noise. In fact, in a reasonably wide window of noise, the system yields logic operations with near certain reliability, i.e., \(P(\text{logic}) \sim 1\). Also notice that, in contrast to the usual peak associated with SR [1], here we have a flat maximum in our performance measure, where the system yields almost 100% accuracy. So the optimal gate performance is robust, as it occurs over a wide range of noise.

Hence, under moderate noise one obtains a large, robust, asymmetric response to input signals: different distinct levels of input pulses can be mapped to a binary 0/1 output, determined by the system being in either one of the two widely separated wells. For instance, for parameter settings yielding NAND logic, the system driven by subthreshold input signals are realized only in the presence of noise. In fact, in a reasonably wide window of noise, the system yields logic operations with near certain reliability, i.e., \(P(\text{logic}) \sim 1\). Also notice that, in contrast to the usual peak associated with SR [1], here we have a flat maximum in our performance measure, where the system yields almost 100% accuracy. So the optimal gate performance is robust, as it occurs over a wide range of noise.


table

<table>
<thead>
<tr>
<th>Input set ((I_1, I_2))</th>
<th>OR (x_1, x_2)</th>
<th>AND (x_1, x_2)</th>
<th>NOR (x_1, x_2)</th>
<th>NAND (x_1, x_2)</th>
</tr>
</thead>
<tbody>
<tr>
<td>((0,0))</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>((0,1)/(1,0))</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>((1,1))</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

The phenomena above can be understood in the standard framework of SR in overdamped systems underpinned by double well potentials [1,2]. The different outputs, obtained by driving the state of the system to one or the other well, are realized by appropriately manipulating the heights and asymmetry of the wells. Straightforwardly extending the earlier analysis one finds that, when inputs \(I_1\) and \(I_2\) are added, the effective position and depth of the wells change to:

\[ x_+ = \left\{ \beta x_0^+ + \varepsilon + I_1 + I_2 \right\} / \alpha \]

and

\[ x_- = \left\{ \beta x_0^- + \varepsilon + I_1 + I_2 \right\} / \alpha \]

This symmetry sends the state of the system to the desired well under adequate noise (see Fig. 4). When \(x_0^+ > |x_0^-|\) one obtains the NOR logic operation, and when \(x_0^- < |x_0^+|\) one obtains the NAND logic operation, with the input sets \((0,1)/(1,0)\) being the distinguishing one for the two types of logic function.

### 3.1. Quantifying reliability

We can quantify the consistency (or reliability) of obtaining a given logic output by calculating the probability of obtaining the desired output for different input sets; this probability, \(P(\text{logic})\), is the ratio of the number of correct logic outputs to the total number of runs. Each run samples over the four input sets \((0,0), (0,1), (1,0), (1,1)\), in different permutations. If the logic output, as obtained from \(x(t)\), matches the logic output in the truth table for all four input sets in the run, it is considered a success; otherwise it is considered a zero success. When \(P(\text{logic})\) is 1 the logic operation is obtained completely reliably, namely the system always yields the correct output. This measure of the success of the logic operation is very stringent, and it checks to see that any random combination of logic inputs, streaming in any random sequence, yields the correct result.

Fig. 5 shows this quantity obtained from numerical simulations over \(10^6\) different runs [6]. It is clearly evident that the fundamental logic operations NAND and NOR are accurately realized in an optimal window of moderate noise. Remarkably, logic outputs for subthreshold input signals are realized only in the presence of noise. In fact, in a reasonably wide window of noise, the system yields logic operations with near certain reliability, i.e., \(P(\text{logic}) \sim 1\). Also notice that, in contrast to the usual peak associated with SR [1], here we have a flat maximum in our performance measure, where the system yields almost 100% accuracy. So the optimal gate performance is robust, as it occurs over a wide range of noise.

Hence, under moderate noise one obtains a large, robust, asymmetric response to input signals: different distinct levels of input pulses can be mapped to a binary 0/1 output, determined by the system being in either one of the two widely separated wells. For instance, for parameter settings yielding NAND logic, the system driven by subthreshold input signals \((0,0)\) and \((0,1)/(1,0)\), goes to the well at \(x\), and input \((1,1)\) results in the system being in well \(x\); whereas for NOR logic, input signals \((0,1)/(1,0)\) and \((1,1)\), send the system to well \(x\), and input \((0,0)\) sends the system to well \(x\).

Now, considering a fixed noise level, one sees from Fig. 6 that different types of logic (NAND vs-a-vis NOR) are obtained in different ranges of thresholds. This can be easily understood from the fact that the thresholds \((x_0^+, x_0^-)\) determine the position, depths and asymmetry of the potential wells, which in turn determine the well the system goes to in response to the input stream. So one can switch between logic functions by simply adjusting a threshold, given a suitable window of noise. This is a very useful feature of the dynamics.

Having introduced a probabilistic measure of “reliability”, it is worthwhile to digress a bit to reiterate that a continuous function characterization of the potential, yields the same phenomena as described above (using the piecewise linear characterization (3)). We have, in fact, considered the dynamics underpinned by a “hard” potential function:

\[ \dot{x} = 2x - 4x^3 + \varepsilon + D\eta(t) + I_1 + I_2 \]

As in the piecewise linear case, interpreting the state \(x < 0\) as logic output 0, and the state \(x > 0\) as logic output 1, we obtain the reliability of obtaining logic operations, \(P(\text{logic})\) (see Fig. 7). It should
come as no surprise that the results conform quite well to those presented earlier for dynamics underpinned by a piecewise linear approximation to a generic (continuous) nonlinearity that would be obtained as the negative gradient of a bistable potential energy function.

3.2. Effect of an applied constant bias signal: a de facto adjustment of the threshold

We have also studied the effect of an additional constant input bias $e$ in (2) (over a temporal interval longer than the noise correlation time). The results are displayed in Figs. 8 and 9. Notice that, as the value of the bias changes (keeping $\alpha$, $\beta$ and $D$ fixed), the response of the system morphs from NOR to NAND logic. The important point here is that changing $e$ changes the symmetry of the potential wells, and this leads to different logical responses.

Fig. 9, obtained through extensive numerical simulations, shows the behavior described in the preceding paragraphs. In each panel we plot the success measure $P(\text{logic})$ vs. the external bias $e$ and the noise intensity $D$. For a given noise intensity (in the optimal range of Fig. 5) this figure shows that adjusting the parameter $e$ will yield the desired logic behavior. Importantly, the noise intensity window at which the performance maxima are obtained in Figs. 6 and 9 overlaps for the NOR and NAND performance. So, when noise intensity is somewhere in the performance plateau, we can switch NOR/NAND operations by simply adjusting the thresholds, or by changing the dc bias $e$.

It is worth noting that, while in this explicit example we have demonstrated 2-input logic, we have also observed multiple input logic behavior. Multiple input logic gates are preferred mainly for reasons of space in circuits and, further, many combinational and sequential logic operations can be realized with these logic gates, in which one can minimize the propagation delay. Such multiple input gates are more power efficient, and have better performance in a wide range of applications. Specifically we have implemented 3-input logic, where the input signal is: $I = I_1 + I_2 + I_3$; namely, the input is a (random) stream of 4-level square pulses.

Before leaving this section, we briefly consider the case of multiplicative noise, inserted via fluctuations in the parameter $a$:

$$\frac{dx}{dt} = -(x + D\eta(t)x + \beta g(x)) + I_1 + I_2$$

$g(x)$ being the piecewise linear function (3). As before, interpreting the state $x < 0$ as logic output 0, and the state $x > 0$ as logic output 1, we obtain the reliability of obtaining logic operations, $P(\text{logic})$, displayed in Fig. 13. So it is evident that LSR occurs for the case of multiplicative noise as well. It is worth noting that, in actual circuits, multiplicative noise is fairly common due to fluctuations in circuit parameters, e.g. resistors. Then again, one
Fig. 4. Timing pulses from top to bottom: panel 1 and 2 show a stream of input signals $I_1$ and $I_2$, with $I_1, I_2 = 0.4$ when logic input is 0 and (ii) $I_1, I_2 = 0.4$ when logic input is 1; panel 3 shows the upper threshold $x_u = -0.4$; panel 4 shows $-x$ giving a dynamic logic response: [0–3000] s is the NAND response, (3000–6000] s is the NOR response.

Fig. 5. (left) Probability of obtaining the NAND logic operation (see text), $P_{(NAND)}$, with $(x_u, x_l) = (-1.3, 0.5)$. A similar result is obtained (right) for $P_{(NOR)}$ taking $(x_u, x_l) = (-0.5, 1.3)$ [3].

Fig. 6. Density map of $P_{(logic)}$ (see text) for the (left) NOR and (right) NAND logic operation, for fixed noise ($D = 0.7$), in the space of upper threshold $x_u$ (x-axis) and lower threshold $x_l$ (y-axis). The light portion of this plot indicates the nonlinearity (determined by $x_u$ and $x_l$ in (2) for which a logic outputs (NOR (left) and NAND (right)) are obtained reliably [3].
Fig. 7. $P_{\text{NOR}}$ (left) and $P_{\text{NAND}}$ (right) as functions of the noise intensity $D$ (x-axis) and asymmetricizing dc input $e$ (y-axis), for the system in Eq. (4). Reliable NAND performance accrues for $e \sim -0.5$; for reliable NOR performance, $e \sim 0.5$. Optimal performance in both cases is obtained for $D$ values lying within the optimal window. $I_i, i = 1, 2$ is equal to 0.5 when logic input is 1, and $I_i, i = 1, 2$ is equal to 0 when logic input is 0.

Fig. 8. From top to bottom: panels 1 and 2 show streams of input $I_1$ and $I_2$, which take value $-0.4$ when logic input is 0 and value 0.4 when logic input is 1; panel 3 shows $e$; panel 4 shows the output of the system, namely the waveform of $-x(t)$ obtained from Simulink simulations of the system given in Eq. (2) (with $x_l = -1, x_u = 1.5$), which can be switched by changing the bias signal $e$. Here we obtain NAND response for [0:5000] s when $e = -0.8$, and NOR response for [5000:10000] s when $e = 0$. Note that the noise level ($D = 2.5$) is in the optimal window here.

Fig. 9. $P_{\text{NOR}}$ (left) and $P_{\text{NAND}}$ (right) as functions of the noise intensity $D$ (x-axis) and asymmetricizing dc input $e$ (y-axis), for the system in Eq. (2). Thresholds are fixed: $(x_l, x_u) = (-0.5, 1.3)$. Reliable NAND performance accrues for $e \sim -1$; for reliable NOR performance, $e \sim 0$. Optimal performance in both cases is obtained for $D$ values lying within the optimal window (Fig. 5). Note that, for $D$ outside the optimal window, one can still, for any $D$, optimize NAND/NOR performance by adjusting $e$ as well as the thresholds $(x_l, x_u)$ (this amounts to an adjustment of the nonlinearity for a given system noise floor) [3].
might wish to introduce controlled multiplicative noise to investigate the response of the circuit and the effect of the noise on gate operation/reliability. This controlled noise could be introduced via a voltage controlled capacitor $C = C_0(1 + Dg(t))$ in place of the constant (linear) capacitor of (2). The multiplicative noise case is, as is well known, one that yields very rich stochastic dynamical behavior, with the potential extrema as well as the energy barrier height depending on the noise intensity parameter[8]. This case is the subject of intense current study.

<table>
<thead>
<tr>
<th>Input bit for number A</th>
<th>Input bit for number B</th>
<th>Carry bit input $C_{in}$</th>
<th>CARRY bit output $C_{out}$</th>
<th>SUM bit output $S$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

### 4. Combinational and sequential logic operations

In this section, we review the implementation details of certain representative combinational and sequential logic systems using concatenation of LSR elements[7]. First we consider a simple full-adder logic operation which involves 3 logic inputs $A$, $B$ and $C_{in}$ (see Table 2 for full truth table). Conventional digital logic circuit for implementing full-adder requires 2 XOR gates, 2 AND gates and 1 OR gate. In the present case, we can use either NOR gate or NAND gate combinations to realize the full-adder logic. As is standard practice, one can realize XOR operation with a set of NOR or NAND gates[4]. Now we configure our LSR element to emulate NAND logic operation by fixing its nonlinearity, and by using a set of LSR elements, XOR logic operation is realized first. Then AND and OR logic operations with LSR are realized easily by inverting their logic output from NAND and NOR operations. Fig. 10 depicts the MATLAB-SIMULINK simulation results for the full-adder results for noise intensity $D = 2$, $x_u^l = 0.5$ and $x_l^u = 1.3$.

Next we consider a more complex sequential logic operation, for example a Data flip-flop operation[4] (see Table 3). This logic operation requires cross-coupling of set of different NOR or NAND logic gates (see Fig. 11 for a schematic). To illustrate the ability of LSR elements to carry out this operation, representative

![Timing sequences from top to bottom for full adder implementation with individual LSR system elements emulating a NAND gate. A set of 4 LSR elements are used to emulate the XOR logic and 2 LSR elements are used for OR logic emulation. In total 14 (2 XOR, 1 OR and 2 AND logic) LSR elements are used to realize the full-adder logic operation (Ref. Table 3): (a) input $A$, (b) input $B$, (c) input $C_{in}$, (d) output $x(t)$ corresponding to SUM, (e) recovered logic output SUM, (f) output $x(t)$ corresponding to CARRY, and (g) recovered logic output CARRY. Here the noise intensity $D = 2.3$, thresholds $(x_u^l, x_u^r) = (1.3, 0.5)$, are used for NAND and thresholds $(x_l^l, x_l^r) = (0.5, 1.3)$ are used for NOR.](image)
5. Experimental implementations

5.1. Analog simulation circuit

The schematic circuit diagram of simulating the dynamics of Eq. (2) is depicted in Fig. 14. This circuit consists of 6 operational amplifiers (Op-amps AD712 or μA741 or AD844), 12 resistors \(R_1 - R_{12}\), 2 diodes \(D_1, D_2\) realized with 1N4148 and 2 variable resistors \(P_1, P_2\). The circuit with op-amp OA1 acts as an integrating amplifier, circuits with op-amps OA2 and OA4 act as sign-changers or inverting amplifiers, circuits with op-amp OA3 act as an inverting summing amplifier, circuit with op-amp OA5 acts as a voltage buffer or unity-gain amplifier and circuit with op-amp OA6 act as a voltage comparator. The node voltages \(V_1, V_2, V_3\) and \(V_4\) are represented as:

\[
\begin{align*}
V_1 &= -\alpha V_2 + \beta V_4 + I_1 + I_2 + D\eta(t) \\
V_2 &= -[-\alpha V_3 + \beta V_4 + I_1 + I_2 + D\eta(t)] \\
\frac{dV_3}{dt} &= -\frac{1}{R_1 C_1} V_2 \\
V_4 &= -V_3
\end{align*}
\]  

MATLAB-SIMULINK results are shown in Fig. 12 in terms of temporal waveforms.
Fig. 13. $P(\text{NOR})$ (left) and $P(\text{NAND})$ (right) as functions of the noise intensity $D$ (x-axis) and asymmetrizing dc input $e$ (y-axis), for the case of multiplicative noise (see (5)), obtained from extensive numerical simulations. Thresholds are fixed: $x_l/C_3/l$; $x_u/C_3/u$; $C_0/C_1/C_24/C_0/C_1$. Reliable NAND performance accrues for $C_0/C_24/C_0/C_1$; for reliable NOR performance, $C_0/C_24/C_0/C_1$.

Optimal performance in both cases is obtained for $D$ values lying within the optimal window. Note that, for $D$ outside the optimal window, one can still, for any $D$, optimize NAND/NOR performance by adjusting $e$ as well as the thresholds $(x_l, x_u)$ (this amounts to an adjustment of the nonlinearity for a given system noise floor).

Fig. 14. Analog simulation circuit of the system represented by Eq. (2). The logic output $V_0$ is measured across the comparator.
on the oscilloscope. i.e., they are represented mathematically as
\[ x = \frac{1}{RC} \frac{dV}{dt} \]

Then, the input signal \( V_i \) is the threshold control voltage
when the logic input is 0, and value 0.5 V when logic input is 1.

Fig. 16. (a) Four transistor cubic-like nonlinear resistor \( N_6 \); (b) Cross coupled 2 inverter implementation of nonlinear resistor \( N_6 \) [9].

Here, \( V_3 \) is the output voltage being generated from the op-amp integrator circuit OA1. The input to this op-amp integrator is
\[ V_3 = -R_1C_1\left(\frac{dV_3}{dt}\right) \]

in terms of summation of voltages as represented in the second relation above. Now, applying the corresponding resistor and capacitor values in accordance with the circuit of Fig. 14, and rescaling the time \( t = R_1C_1t \), we get (2) provided the parameters are redefined as \( \tau = t \), \( P_2 = \beta \), and \( P_1 = \alpha \). Also all the node voltages \( V_i (i = 1 \ldots 4) \) are observed with respect to time on the oscilloscope. i.e., they are represented mathematically as \( V_i(t) \).

We set the values \( R_1 = R_6 = 90 \), \( R_3 = 14K \), \( R_2 = R_3 = 3 \), \( R_7 = 8 \), \( R_10 = R_11 = R_12 = R_13 = 100K \), \( C_1 = 10 \) nF, redefining \( V_3 \) as \(-x(t)\), and \( V_4 \) as \( x(t) \). Choosing \( x = 1.8 \) and \( \beta = 3.0 \), the circuit will give the waveform \( x(t) \), \( x(t) \) and logic output waveform \( V_0 \) from the comparator with threshold reference value \( x^* = 0 \) V.

In actual experiments, \( D \) is the noise intensity denoted in terms of voltage and the noise is generated by using a noise function generator (e.g., Agilent HP33120A). With no loss of generality, consider the two (randomly switched) inputs \( I_1 \) and \( I_2 \) to take values \(-0.5 \) V when the logic input is 0, and value 0.5 V when logic input is 1. Then, the input signal \( I = I_1 + I_2 \) is a three-level aperiodic square waveform. Further, by considering the diode cut-off voltage (in the present case, 0.7 V), the threshold control voltages \( V_0 \) and \( V_1 \) are suitably fixed so that the effective threshold levels will be set at desired values of \( x^* \) and \( x^* \). Fig. 15 shows the experimental trace of the system morphing between NOR and NAND logic output, by a change in one of the threshold settings.

5.2. CMOS realization of a LSR circuit

LSR is demonstrated in this section via a circuit implementation using a linear resistor, a linear capacitor and four CMOS-transistors with a battery to produce a cubic-like nonlinearity. This circuit is simple, robust, and capable of operating in very high frequency regimes; further, its ease of implementation with integrated circuits and nanoelectronic devices should prove very useful in the context of reliable logic gate implementation in the presence of circuit noise.

The simple CMOS based circuit we have proposed in [9], has very few circuit elements. It consists of a linear resistor \( R \), linear capacitor \( C \), and a nonlinear (constructed to be cubic) circuit element (Fig. 1, right panel) implemented using CMOS transistors. The circuit can be visualized as a pair of cross-coupled CMOS inverters (CD4007CN) as shown in Fig. 16. The subcircuit is characterized by low parasitic capacitance and, when used as a cubic-nonlinear element, the resulting circuit can operate across a very high frequency range [10]. We construct this element with a Dual Complementary Pair Plus Inverter Chip [10] powered by single 9 V battery. To achieve maximum speed of operation using this nonlinearity, the parasitic capacitance across its terminals was used as the circuit element \( (C = 49 \text{ pF}) \) in the LSR circuit of Fig. 1 (right panel). The governing circuit equation is
\[ RC \frac{dV}{dt} = -V - R(V) + f(t) \]  
(7)

where \( f(t) = Dn(t) + h_1 + l_2 + c \), and \( g(V) \) is the driving-point characteristic of the nonlinear resistor represented as a smooth cubic nonlinearity: \( g(V) = G_0 V - \frac{G_0}{2} V^3 \), with \( G_0 \) being the (experimentally measured) slope, at the origin, of the nonlinearity. The change to dimensionless variables \( \tilde{x} = x \tilde{V} \), \( \tilde{t} = \frac{t}{\tilde{V} \tilde{C}} \), \( a = \frac{\tilde{V} \tilde{C}}{\tilde{G}_0} \) leads to an equation that is convenient for analysis and numerical simulation:
\[ \tilde{x} = f(\tilde{t}) - Ax + A\tilde{x}^3. \]  
(8)
with the definitions \( A = 1 + a \) and \( f(t) = I_1 + I_2 + \varepsilon + D\eta(t) \). We note that (8) is the prototype overdamped system with dynamics derived from the bistable quartic potential \( U(x) = A\left(\frac{x^2}{2} - \frac{x^4}{4}\right) \); note that \( a \) and \( A \) are both negative. This system exhibits stochastic resonance for the appropriate choices of system and noise parameters [1]. Absent the forcing function \( f(t) \), the switching between the stable states \( x_s = \pm 1 \) is, solely, noise-driven if \( D \neq 0 \).

In Fig. 17 we show the response of the system (7) for \( \varepsilon = 0.2 \) V, \( a = -2.75 \), and three choices of noise intensity \( D \). We observe that, under optimal noise, interpreting the capacitor voltage \( V < 0 \) as logic output 0 and the voltage \( V > 0 \) as logic output 1 yields a robust logical OR. In a complementary fashion, interpreting the capacitor voltage \( V > 0 \) as logic output 0 and \( V < 0 \) as logic output 1, yields a logical NOR. In a completely analogous way, by setting the asymmetry value \( \varepsilon = -0.2 \) V, we can realize clearly defined AND and NAND gates in almost the same optimal noise intensity regime as the previous case.

6. Conclusions

In summary, we have shown how the constructive interplay between a noise-floor and nonlinearity can be exploited to obtain extremely reliable logic gate output from a bistable system. Additionally, we have shown how the system can also switch the logic response by adjusting the nonlinearity or bias parameters, for a given value of the noise-floor. Put differently, this adjustment allows us to effectively manipulate the (nonlinear) transfer characteristic of the system to optimize the logic response for a given noise-floor. So one can use the nonlinearity (or the constant bias) to robustly select different logic truth tables. So a major implication of these observations is the realization that such systems can work as robust reconfigurable logic gates [11].

Finally, it is conceivable that LSR may be observed even in the quantum realm [12]. Very recently, the feasibility of LSR in nanomechanical oscillators [13], and in chemical systems [14] has been demonstrated. Such results underscore the broad reach of the concept of LSR, and opens new directions in noise-assisted computing devices.

Acknowledgment

KM, WLD, AD, and ARB were supported by the Office of Naval Research through Grant Nos. N000140211019 and 000148WX20330AA.

References

[5] Simulink is a simulation and model-based design tool used extensively in engineering applications.
[6] Numerical simulations of the above stochastic differential equation were done by the standard Euler method that converges to the Ito calculus.