

A Memristive Switching Uncertainty Model

Spyros Stathopoulos, Alexantrou Serb, *Member, IEEE*, Ali Khiat, Maciej Ogorzałek, *Fellow, IEEE* and Themis Prodromakis, *Senior Member, IEEE*

Abstract—In this paper we endeavour to evaluate and model switching noise in resistive random access memory devices (RRAM). Although noise is always present in physical systems, the sources of which can be attributed to many different effects, in this study we are focusing our attention on a specific type – switching noise. Using alternating pulse programming and read trains across different voltages we acquire a large dataset below and above the switching threshold and construct what we define as *increment plots*, ΔR vs. R . Then, through a detailed statistical analysis, we quantify the localised uncertainty among consecutive points using a sliding window of up to N points accounting for any statistical artefacts that arise. By separating the data accumulated from programming and read-out and analysing them individually we can subtract a baseline noise floor from the overall switching uncertainty. In this way we effectively decouple it from other noise sources that affect the device at rest. In the end an $F(R, V)$ surface can be extracted that closely follows the behaviour of uncertainty of the device during programming. This modelled surface can be used as an approximation of the noise behaviour of the device or it can be readily incorporated as an additional component to existing switching models.

Index Terms—memristor, RRAM, switching noise

I. INTRODUCTION

THE correlation of the *memristor*, the theorised fourth passive circuit element [1], with *resistive memories* as realised by Strukov et al. [2], has received remarkable attention due to the broad range of potential applications that have been put forward some of which include, but are not limited to, non-volatile multi-bit memories [3], neuromorphic systems [4] and reconfigurable circuits [5]. Resistive memories, however, are not immune to the effects of noise [6] which is present in all physical systems whether it is internal or external to them [7]. In certain cases [8] memristive response can be enhanced by the presence of noise.

Manifestations of noise attributed to telegraph noise (RTN) and random walk have indeed been researched before in the context of resistive memories [9–11]. In [12] the authors present a way to model RTN in RRAM devices with highly quantised resistive states as well translate its implications to circuit design. Noise effects were also taken into consideration

as part of more holistic bottoms-up physical/chemical RRAM models dealing with specific technologies [13, 14]. Indicative sources of uncertainty in memristive systems can be due to *cycle-to-cycle* variability, *switching rate* uncertainty or *instrumentation noise*. This, of course, is not an exhaustive list as additional factors such as drift due to device ageing, Johnson-Nyquist noise, $1/f$ noise and possibly others can affect the device at rest. In Figure 1 a memristive device is under repeated pulsed programming bias. The overall trend of the resistance is increasing, so the device is indeed switching to a different resistive level; there is, however, a degree of uncertainty in-between consecutive data points. This uncertainty is expressed in a form of noise during the progress of the measurement. Since the uncertainty introduced is an aggregate of many different sources there cannot readily be a distinction between the uncertainty due to random effects and noise associated with purely *switching* the devices to different resistive levels, which is in effect what we call the *switching noise*. RTN-like effects are considered as an uncertainty during read-out or in general when the resistance of the device is assessed. Switching noise is, instead, the manifestation of uncertainty atop a specific phenomenon and occurs during programming.

This is a key parameter to optimise when constructing behavioural models for memristive devices, especially with devices exhibiting gradual switching between close neighbouring states, and allows for noise components to be added to existing models closely matching the behaviour of real devices within the constraints of the operational resistive range. This point is especially pertinent in the case of engineering neuromorphic computing systems as every plasticity event can change the synaptic weights. That weight dispersion will affect the trustworthiness and reliability of the system [15].

Towards that end, this work aims to address this issue by providing a characterisation methodology and an accompanying statistical analysis to model the switching noise while decoupling it from underlying non-switching components, as those that arise during read-out when the device is at rest. Instead of treating noise a global source of variability we endeavour to isolate its effect during the programming phase. Our proposed model can be then readily incorporated to existing behavioural switching models without further modification of the original.

This paper is organised as follows. In section II we present the methodological background to the paper in order to properly estimate switching noise. Section III deals with the actual experimental translation of the methodology. In section IV we present and discuss upon the experimental results. Concluding remarks follow in section V.

This work has been supported by the EPSRC EP/R024642 and EP/K017829/1 programme grants and the National Science Centre of Poland grant 2015/17/B/ST7/03763.

S. Stathopoulos, A. Serb, A. Khiat and T. Prodromakis are with the Electronic Materials and Devices Group, Zepler Institute for Photonics and Nanoelectronics, University of Southampton, SO17 1BJ, Southampton, UK (e-mail: {s.stathopoulos, a.serb, a.khiat, t.prodromakis}@soton.ac.uk).

M. Ogorzałek is with the Department of Information Technologies, Jagiellonian University, 30–348, Krakow, Poland (e-mail: maciej.ogorzalek@uj.edu.pl).

Data for this paper is available from the University of Southampton institutional repository at <https://doi.org/10.5258/SOTON/D0929>

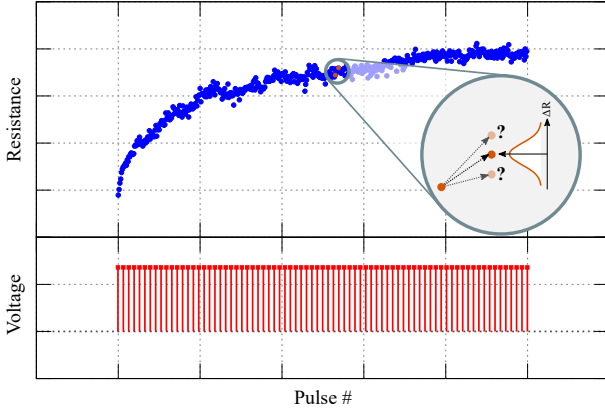


Fig. 1. Resistive response of a device under constant pulsed bias. Although resistance is generally trending upwards there is some degree of uncertainty regarding the relative change between consecutive data points. This is manifested as switching noise which is distinct from other forms of noise. Units are arbitrary.

II. METHODOLOGY

We can define *switching* as a change in resistive state (RS), measured under a fixed read-out voltage, before and after the application of a stimulus of a specific combination of amplitude and pulse width [16]. In an ideal voltage controlled memristive system as described in [17], if any other factors are ignored, the degree of switching should in principle only depend on initial RS (R) and applied stimulus waveform (in this case, voltage amplitude of square wave stimulation V). However a number of uncertainty factors mean that change in resistance, ΔR , will typically differ for each trial. We call this phenomenon *switching uncertainty* or *switching noise*.

The overall flow of the switching noise estimation process is as follows: In order to assess the degree of switching noise we must sample the degree of resistive switching from multiple initial resistive states, for multiple voltage stimulation magnitudes. To that end we employ an automated stimulus generation protocol for data collection. Next, we note that the ΔR depends on the last measured resistance, R , given an underlying ground truth value of R_0 and compensate for that effect. Then, we estimate the switching uncertainty locally at each point on the R - V plane (our input space consisting of the resistive state of the device and applied voltage amplitude). Finally, all locally gathered data is aggregated into a surface describing switching uncertainty as a function of R and V .

A. Data acquisition and preparation

In this work the basic unit of device stimulation consists of a pair of square wave voltage pulses. First a programming pulse is applied. This may have a varying amplitude but fixed duration as shown in Figure 2a. Next, a low amplitude (below the threshold that causes the memristor to switch) read-out pulse is used to measure the resistive state of the device under standard voltage conditions. Of course the selection of the read voltage is technology dependent as, in principle, long enough biasing even at low voltages can eventually lead to a switching event.

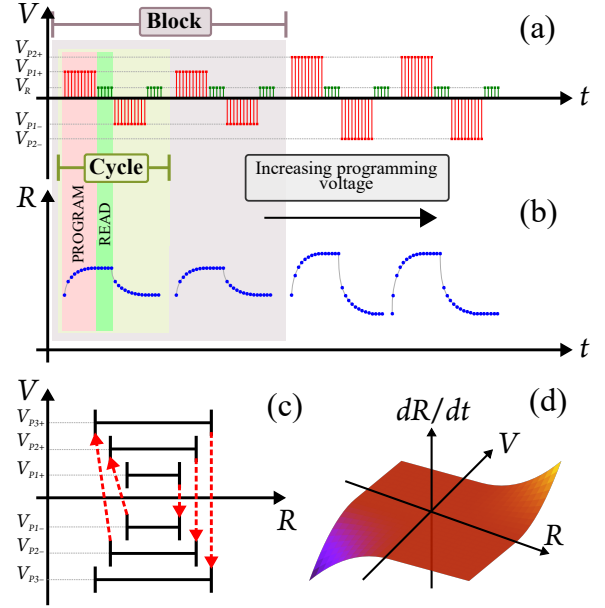


Fig. 2. Stimulus protocol as described in paragraph II-A. A series of alternating polarity pulse trains (*cycle*) is applied a predefined number of times forming a *block* of stimulus at a specified voltage (a & b). The voltage of the block is increased to cover a large area in the R - V plane (c). The switching rate of the device as a function of R and V can be estimated (d). Read-out trains (in green in Fig. a) are applied in-between the programming trains as a means to extract the baseline noise of the device.

Our stimulation protocol is a variation of that demonstrated in [18]. Pulsed voltage trains of alternating polarities and progressively increasing voltage are used to induce switching in a test memristor. Each pulse train causes the device to saturate at a specific voltage-dependent RS (Figure 2b). Each pair of bipolar pulse trains acts as a *switching cycle*, jogging the resistive state of the device towards higher (or lower) resistances and back. Many cycles using the same voltage amplitudes can be chained together to form a *block*. The block voltage amplitudes are progressively increased up to a maximum value thus covering an increasingly larger area in the R - V plane (Figure 2c). The end result is an extracted model which describes the normalised switching rate of the device (dR/dt) as a function of initial resistance and bias voltage as in Figure 2d. After each programming pulse a series of successive read-outs is carried out in order to assess the baseline noise of the device when at rest at a specific resistive level.

After the above stimulation protocol has been applied the resistances that have been accumulated are differentiated, separately for programming and read-out pulses, and the *increment plots*, ΔR vs. R , are generated. Increment plot data (R , ΔR) corresponding to equal voltages within each block is pooled together for the purposes of the analysis carried out in the following sections.

B. Correlation compensation

Increment plots are useful for illustrating the dependence of resistive state change on the running value of resistance. At

this point it is important to introduce the concept of *underlying* resistance. In a system without any noise the resistance of the device would either be stable (under read-out) or change monotonically (under programming). This ground truth is what we define as *underlying* resistance for the purposes of this paper. Due to the effect of noise *measured* resistance includes a perturbation around the *underlying* one which, as a result, is never directly known during the course of an experiment.

Calculating the uncertainty from a increment plot is straightforward when the underlying resistive state does not change, as in during read-out of a device. An illustrative example with generated data can be seen in Figure 3a. The corresponding increment plot for this particular case is shown in Figure 3c. In order to extract a histogram of the ΔR values corresponding to our specific underlying R_0 from the increment plot we merely need to ignore the absolute R values and extract the histogram of the ΔR values. We may do this because it is clear that all data points in the increment plot correspond to the underlying value R_0 .

Investigating Figure 3c in more detail we notice that the data points of the increment plot are forming an elongated cluster centred around $\Delta R = 0$ and $R = R_0$. The cluster is noticeably rotated by 45° . The reasoning is as follows: If we sample a random variable, x , drawn from a normal distribution with mean μ and standard deviation σ then the expected value for each sample would be, by definition, μ . Therefore if we draw a value with offset k from the mean, $S(t) = \mu + k$, the expected difference, $\langle \Delta S \rangle$, will then be:

$$\langle \Delta S \rangle = \langle S(t+1) - S(t) \rangle = \mu - (\mu + k) = -k \quad (1)$$

It follows that in the case of Figures 3a and c for the resistive values of a device:

$$\langle \Delta R \rangle = \langle R(t+1) \rangle - R(t) = R_0 - R(t) \quad (2)$$

the slope, $\partial \Delta R / \partial R$, will on average be -1 i.e. lying on a line 45° against the y -axis as shown in Figure 3c.

In the case of a time variant underlying resistive state (Figure 3b) uncertainty extraction from the increment plot becomes more involved. For a device that changes its resistance under bias (as in during programming), the data points shown in Figure 3d will, in general, correspond to *different* values of underlying resistance. However, isolating a number of *time-consecutive* data points in the increment plot 3d we can observe that the same pattern as in 3c emerges as each cluster of arbitrary consecutive points (highlighted in Figure 3d) lies on a 45° orientation. In other words, the time-varying case can be understood as (approximately) producing a succession of time-invariant clusters around different underlying (mean) resistances.

In order to compensate for this 45° rotation effect we may apply a standard -45° rotation matrix across all data, both in the high and low resistive states. The results of this rotation process are shown in Figures 3(e,f) for the time-invariant and time-varying case respectively. Now data points corresponding to the *same* underlying value of R are vertically aligned to the best approximation as it is evident from the highlighted subcluster. This simplifies the execution of the next steps in the process.

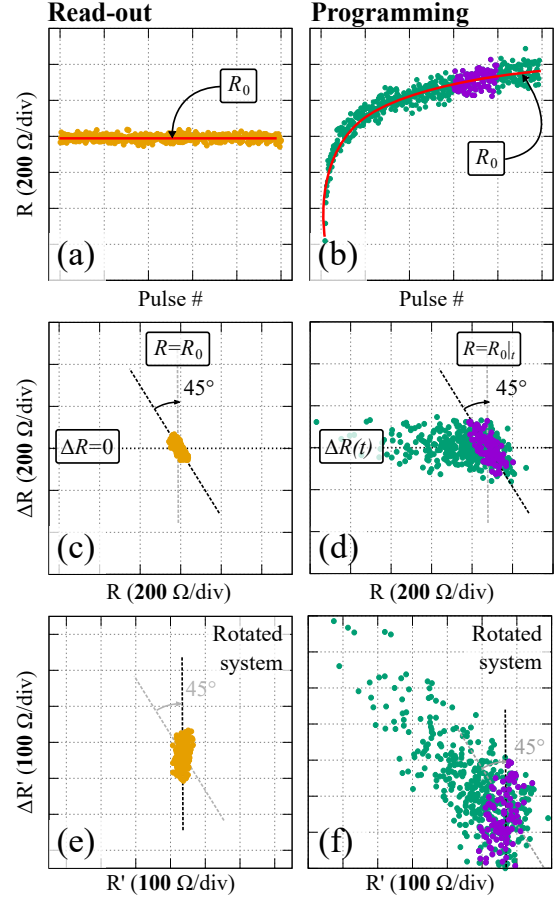


Fig. 3. Increment plot construction and rotation for a set of randomly generated (i.e. not measured) data around an *invariant* value (a) or a *time-variant* value (b). The underlying unperturbed resistance R_0 is indicated for both cases. During the course of an experiment R_0 is typically unknown. For the invariant value plotting the increment plot ΔR vs R leads to the data points being arranged randomly around an axis lying 45° with respect to the y -axis (c). For the case of the time-variant value the same observation approximately holds for *each subcluster* of arbitrary but *time-consecutive* data points. Values are arbitrary; scales for all axes in each row are identical; aspect ratio for all subfigures is 1:1; labels indicate the number of units per division per axis. Applying a -45° rotation transformation matrix leads to the data points being aligned vertically (e,f). A subcluster of data (in purple) is isolated to illustrate the effect of the transformation on the dataset.

C. Uncertainty estimation and statistical correction

Having vertically aligned data points estimated to correspond to the same underlying RS we may proceed to calculate the level of uncertainty as a function of RS. To that end we employ a *sliding window* of N data points in the rotated $\Delta R' - R'$ plane. In our case we chose $N = 3$. Each of these *triplets* of points is consecutive along the R' axis (the x -axis in the rotated data shown in Figure 3). A highly localised, even if rough, estimate of uncertainty can be extracted for each triplet.

An indicative triplet based on the randomly generated data of Figure 3 can be seen in Figure 4a. We notice that the points are distributed both along the y -axis and the x -axis. Performing an N -point *Gaussian fit* along the y -axis yields: a) a *mean* value, μ_y , linked to the estimated “true” relative switching difference and b) a *variance*, σ_y , linked to the magnitude of switching noise. The average of the x -axis

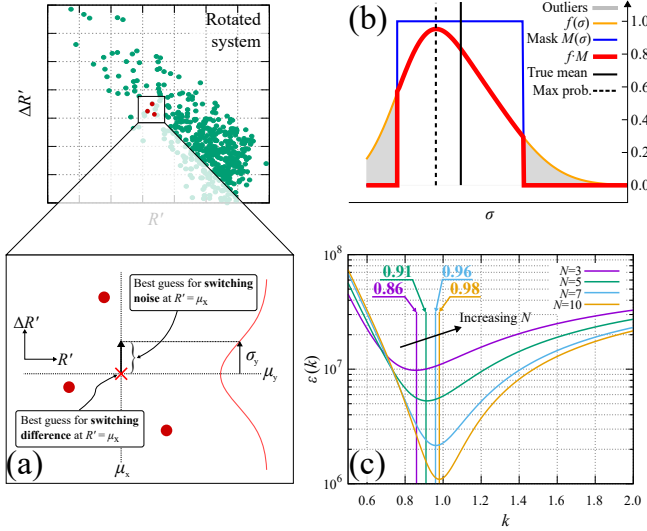


Fig. 4. Uncertainty estimation and statistical correction. (a) Localised calculation of average uncertainty among a triplet ($N = 3$) of consecutive points on the $\Delta R' - R'$ plane. (b) Distribution of σ s calculated triplet-wise from normally sampled data. The most probable value (dashed line) of the resulting distribution lies away from the true mean of the original data (solid line). The quadratic error function, $E(\sigma)$, is applied on the distribution which is then additionally masked by the window function $M(x)$ resulting in the final masked function (in red). (c) Integral of the $f(k\sigma)M(k\sigma)E(\sigma)$ error function for various values of k and N . As N increases the value of k that minimises the error function trends asymptotically to 1, i.e. the most probable value of the distribution lies on the “true” mean of the original dataset.

coordinate, on the other hand, μ_x , yields an estimate of the location along the R' axis to which the data from the y -axis fit corresponds.

μ_y is an estimator of the “true” relative switching difference calculated to accommodate the available data points in the triplet i.e. minimises the estimate for σ_y . As a result triplet-wise calculations of switching noise σ_y underestimate the true magnitude of the noise. An example is illustrated in Figure 4b where we see the distribution of σ s as extracted triplet-wise from a series of normally-generated data with fixed underlying value. We notice that the maximum probability density point (highlighted as a dashed vertical line in Figure 4b) is substantially different from the actual variance of the original data (highlighted as a solid vertical line in Figure 4b). This in itself is not problematic however it hints towards the possibility that using the values for σ_y as extracted might be sub-optimal.

In order to obtain more useful estimates of σ_y we may stretch the distribution of σ_y s along the x -axis by a factor of k i.e. perform the transformation $f : f(\sigma) \rightarrow f(k\sigma)$. Next we need an error function, $E(\sigma)$, to quantify the discrepancy between the distribution of σ s, $f(\sigma)$, and the true underlying value, σ_r . The choice of error function is free but in this work we use the standard square error function $E(\sigma) = (\sigma - \sigma_r)^2$. Finally we may choose to further apply a masking function, $M(\sigma)$, upon the distribution if we wish to exclude the outlying tails from the error calculation. In this work we choose $M(\sigma)$ such as to exclude the top and bottom 10th percentiles of the data. The masking function allows us to calculate our correction factors so that larger number of the estimated σ s lie closer to the true value. Naturally, any transformation applied

TABLE I
STEPS TO EXTRACT MODELLED NOISE SURFACE $N(R, V)$

Step #	Description	Input	Output
<i>For each voltage</i>			
1 (II-A)	Data acquisition	—	R
2 (II-A)	Data differentiation	R	$(R, \Delta R)$
3 (II-B)	Transform $R \rightarrow R'(45^\circ)$	$(R, \Delta R)$	$(R', \Delta R')$
4 (II-C)	N -wise uncertainty calc.	$(R', \Delta R')$	$\sigma(R')$
5 (II-C)	Statistical correction	$\sigma(R')$	$k, \sigma_{\text{cor}}(R')$
6 (II-C)	Transform $R' \rightarrow R(-45^\circ)$	$\sigma_{\text{cor}}(R')$	$\sigma_{\text{cor}}(R)$
<i>After all voltages have been processed</i>			
7 (II-D)	Surface fittings	$\sigma_{\text{cor}}(R, V)$	F, B
8 (II-D)	Background removal	F, B	$N(R, V)$

to $f(\sigma)$ must be matched by an equivalent transformation on $M(\sigma)$. As a result our error metric, ϵ , becomes

$$\epsilon(k) = \int f(k\sigma)M(k\sigma)E(\sigma)d\sigma \quad (3)$$

The function $f(k\sigma)M(k\sigma)$ is highlighted in red in Figure 4b. The next step is to find the value of k that minimises $\epsilon(k)$. The optimal value of k will depend on the value of N so it will be different for triplets, quintuplets, etc. and tends asymptotically towards 1 for increasing N . In Figure 4c error metrics, $\epsilon(k)$ are plotted for different values of N . When N is equal to the total sample size the original and corrected distributions will be identical. For the case illustrated in this paper ($N = 3$) we get that the error metric is minimised for $k = 0.86$.

At this point we apply the $k \cdot 1/\sqrt{2}$ factor correction on our triplet-wise values of σ to obtain an adjusted estimate of switching noise, σ_{cor} corresponding to the value of R' indicated by μ_x (in the $R' - \Delta R'$ space). The $1/\sqrt{2}$ factor is necessary because σ_{cor} is an estimate of uncertainty based on the distance metric running in the -45° rotated direction as can be observed in Figure 3c. In the same Figure we notice that only the uncertainty in ΔR (parallel to the ΔR axis) is relevant. Finally, rotating back the points $\sigma_{\text{cor}}(R')$ by 45° we obtain the function $\sigma_{\text{cor}}(R)$, i.e. localised levels of switching noise vs. running resistance, R .

It should be noted that the choice of N is ultimately grounded in the fundamental problem of estimating a variable that is simultaneously the time derivative of one of the influencing parameters. In a sense the N -wise calculation acts a “smoothing filter” on top of the existing data. Opting for larger N would of course provide a better estimation for the Gaussian but would otherwise dilute the locality of the data. In the end this is a trade-off that must be taken into consideration when modelling a specific device technology.

D. Switching surface and model construction

Performing the analysis outlined in sections II-B and II-C across the different voltages present throughout all blocks in the test (also see Figure 2c) yields an overview of $\sigma_{\text{cor}}(R)$ across a range of bias voltages $\sigma_{\text{cor}}(R, V)$. This can then be fitted to an appropriate surface, $F(R, V)$ consisting of two

component surfaces; one for each voltage polarity, $F_+(R, V)$ and $F_-(R, V)$. The plane fit can be performed with different methods although in this paper we are using multi-variable least squares.

Furthermore, we may now process the read-out data as shown in Figure 3(a, c and e) which yields an estimate of residual (*non-switching noise*-related) uncertainty at the various resistance levels, R , where we have performed the reading operations. Because we always perform the read-out at a fixed voltage residual uncertainty under the read operation is assumed independent of *programming* pulse voltage. This can still be described as a plane in the R - V space, $B(R, V)$.

The final step is to now remove this residual uncertainty from the estimation of switching noise. Since $F(R, V)$ is constructed from data where switching noise is the dominant factor, i.e. much higher than $B(R, V)$ (the read-out “floor”). Plane $F(R, V)$ is then fitted to that points and extrapolated. As long as $F(R, V)$ is not parallel to $B(R, V)$ the extrapolated noise level is guaranteed to fall below the baseline read-out level. Continuing to extrapolate past that point has clearly no physical meaning. We also assume Gaussian distributions for both switching noise and residual uncertainty. Therefore, under these assumptions, the switching noise surface, $N(R, V)$, can be given by the equation.

$$N(R, V) = \begin{cases} \sqrt{F^2(R, V) - B^2(R, V)} & F > B \\ 0 & \text{otherwise} \end{cases} \quad (4)$$

Our proposed noise surface estimation methodology is summarised in Table I.

$N(R, V)$ is the final switching noise model of the device. This can be readily introduced as an additional module to the model described in equation 5 of [17]. The core model functions by approximating arbitrary input waveforms as sequences of suitable short fixed voltage pulses and estimating δR after each pulse. Therefore, in order to incorporate the noise model we must be able to estimate $N(R, V)$ for pulses of arbitrary duration based on data extracted using some chosen fixed pulse duration. To that end, assuming noise behaviour under many short pulses versus few long pulses of fixed *aggregate* duration is the same we may use the Bienaymé formula to calculate expected switching for any duration of pulse. An important point to be made here is that V in the $N(R, V)$ expression is the voltage bias used to *induce the respective resistive change* rather the voltage used for read-out.

For example, if we want to estimate $N(R, V)$ for pulsed duration τ and we know $N(R, V)$ for pulse duration T we may express $\tau = aq$ and $T = bq$ where q is a suitably small time duration quantum. By Bienaymé

$$\sigma_\tau^2 = a\sigma_q^2 \quad \text{and} \quad \sigma_T^2 = b\sigma_q^2 \quad (5)$$

and therefore it is easy to prove that

$$\sigma_\tau = \sqrt{\frac{a}{b}} \sigma_T \quad (6)$$

Therefore if $N(R, V)$ is stored into the model for some known T then every time the model needs to calculate a δR we may draw a random variable out of Gaussian distribution centred

at the predicted noiseless δR with standard deviation derived based on time-corrected $N(R, V)$.

One particular advantage that our approach presents is the versatility of the modelled surface. In any model that accounts for uncertainty the noisy factor should be developed and introduced into the parameters of the system. This can be either a perturbation on an existing parameter or the introduction of a new factor that does not directly translate to an existing one. Ultimately, if the uncertainty captured by one model needs to be transferred to another then it must be translated into the terms of the new one. However, because of our data-driven approach we can simply implant a new uncertainty factor without interacting, in principle, with any pre-existing parameter of the “importing” model.

III. IMPLEMENTATION AND DEVICES

Algorithm 1 Pseudocode for the characterisation protocol in this paper (as described in section III)

```

1: Globals  $V_{\min}, V_s, V_{\max}, C, N_p, N_r \triangleright$  as defined in Table II
2:  $V \leftarrow V_{\min}$   $\triangleright$  initial voltage
3: while  $|V| \leq |V_{\max}|$  do
4:    $i \leftarrow 0$   $\triangleright$  cycle counter
5:   while  $i < C$  do
6:     PROGRAMTRAIN( $|V|, N_p$ )
7:     READTRAIN( $N_r$ )
8:     PROGRAMTRAIN( $-|V|, N_p$ )  $\triangleright$  inverse polarity
9:     READTRAIN( $N_r$ )
10:     $i \leftarrow i + 1$ 
11:   end while
12:    $V \leftarrow V + V_s$   $\triangleright$  increase voltage
13: end while

```

The methodology outlined in this paper has been implemented on top of our in-house characterisation platform as in [19] using the protocol outlined in Figures 2(a,b) and described in section II-A. The elementary stimulus unit, which we call a *cycle*, is a pair of alternating polarity programming trains of a predefined amplitude, (starting with V_{\min}), pulse width (τ_p) and interpulse interval ($\delta\tau$). A fixed number (N_r) of read-out pulses are applied between each polarity reversal. This process is repeated C times until the current block is

TABLE II
PARAMETERS USED IN THE CHARACTERISATION ROUTINE

Parameter	Description	Value	Unit
V_r	Read-out voltage	0.2	V
V_{\min}	Initial programming voltage	Variable	V
V_s	Programming voltage step	0.1	V
V_{\max}	Max programming voltage	Variable	V
τ_p	Programming pulse width	1.0	μ s
$\delta\tau$	Interpulse interval (prog. and read)	1.0	ms
N_p	Number of programming pulses	500	N/A
N_r	Number of read-out pulses	150	N/A
C	Cycles per block	3	N/A

completed. Then the voltage is increased using a fixed voltage step, V_s , until the maximum voltage, V_{\max} is reached. Initial and maximum voltages are device-dependent while the rest of the parameters have been fixed to the values shown in Table II. The pseudocode outlining the stimulus protocol can be found in listing Algorithm 1.

As a specific *case study*, we applied the methodology and analysis expounded in this paper upon a set of Pt/TiO₂/Al₂O₃/Pt devices (layers are from bottom to top) fabricated on 6-inch oxidised silicon wafer. The SiO₂ was grown thermally with dry oxidation and is 300 nm thick. The top and bottom platinum electrodes are 10 nm thick fabricated using e-gun evaporation. The active layers TiO₂ and Al₂O₃ are 25 nm and 4 nm thick respectively and are deposited using the reactive magnetron sputtering technique. Devices with such configuration have been shown to retain a multitude of stable resistive states [3] and are therefore well-suited for our testing purposes. Over 30 devices have been measured and for the purposes of this paper eight representative test cases have been isolated covering resistive range from the low-k Ω to low-M Ω range (DUT 1 to DUT 8).

Before testing, devices have been electroformed from their pristine state using negative polarity voltage ramps of 10 μ s pulse width and amplitudes increasing from 7 up to 11 V with a 0.25 V step. This procedure brings the resistance of the devices to the \sim 500 k Ω range and is similar to what we have used in previous papers [3]. At this point devices exhibit varying degrees of volatility. In order to properly assess noise we require our devices to be devoid of additional volatility effects. So a further electroforming step with both varying pulse width of 0.1 up to 10 μ s and varying amplitude (2–7 V) brings the device to their working non-volatile range. For the purposes of this demonstration the read-out voltage was set to 0.2 V (up to 50 ms pulse width) while the programming voltage was varied with pulse width set at 1 μ s. Short pulses are used to limit current overshoot without current compliance. Pre-determined waveforms are applied on the device and the overall energy is limited by the short time-scales involved.

IV. RESULTS AND DISCUSSION

Once the devices have been electroformed to their initial resistances the characterisation routine outlined in this section is applied. Initial and maximum voltages for each of the devices tested as well fitted parameters are summarised in Table III. Noise model extraction is done separately for each of the positive and negative branches of the accumulated data as a device is not necessarily symmetrical with respect to the applied voltage polarity.

Figure 5a illustrates a block of three testing cycles with 500 programming and 150 read pulses at 1.9 V and -1.9 V programming voltage for DUT 2. The programming pulses alternate the device between two neighbouring resistive states (\sim 7.3 and \sim 8.1 k Ω). Read pulses are applied between any bias polarity change and are used to assess the baseline noise (i.e. not attributed purely to switching). From the acquired data the increment plots 5(b,c) are constructed as per sections II-A and II-B separately for the programming and read pulse trains.

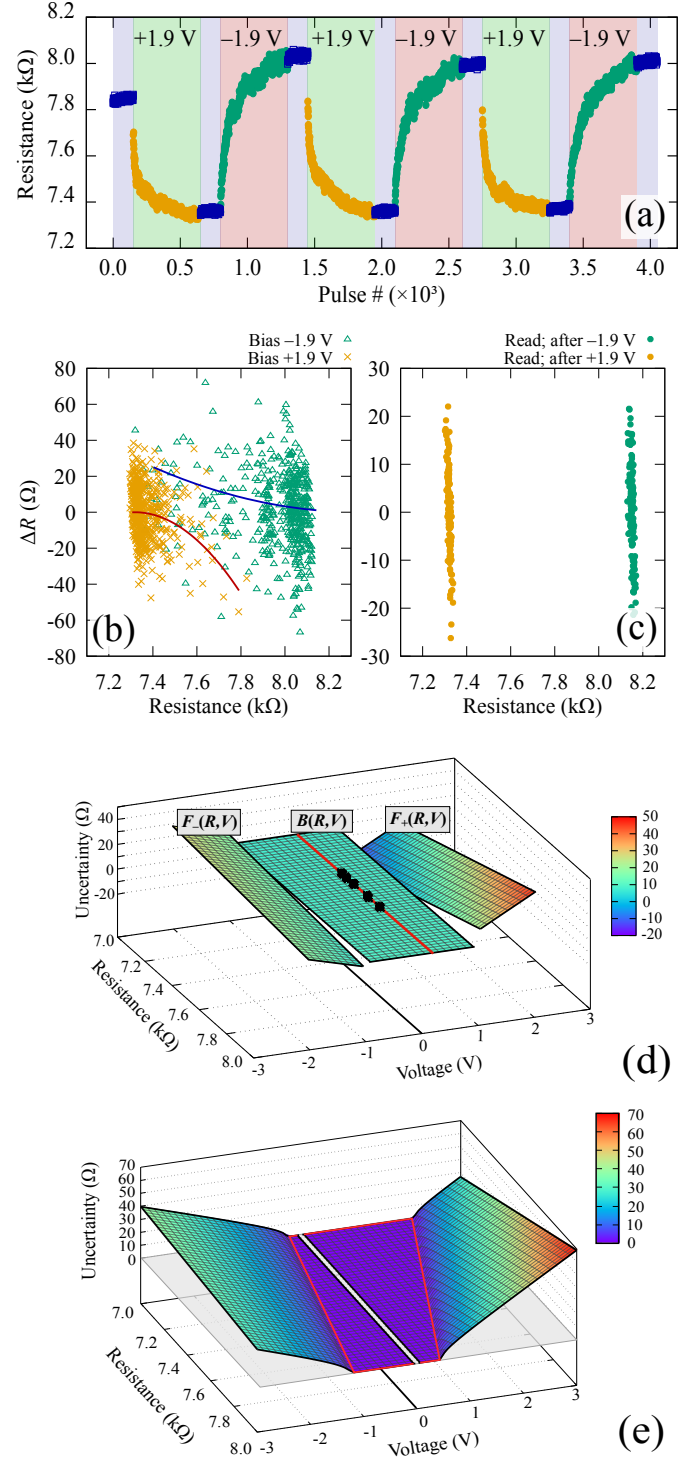


Fig. 5. Switching noise extraction for one of the tested devices (DUT 2). (a) Response of the device during one block of three switching cycles at \pm 1.9 V. In-between each programming train of 500 pulses a series of 150 read pulses is applied; increment plots extracted from the programming (b) and read-out (c) phases directly after; (d) Switching uncertainty planes for read-out, $B(R, V)$, and programming, $F_{\pm}(R, V)$. Planes are fitted across all distributions of σ_s for each voltage for $V > 0$, $V < 0$ read-out data (first-order approximation). The black points at the read-out voltage (0.2 V) are used to extrapolate the baseline plane $B(R, V)$ (e) Switching uncertainty surface, $N(R, V) = \sqrt{F^2(R, V) - B^2(R, V)}$, in a first-order approximation after extracting background uncertainty as per eq. 4. In the highlighted area calculated noise is below the read-out noise threshold ($F < B$) and cannot be attributed purely to switching (and is therefore clipped to 0). Trend-lines in (b) are to guide the eye only.

TABLE III
RESULTS FOR TESTED DEVICES ACROSS DIFFERENT RESISTIVE RANGES

	DUT 1		DUT 2		DUT 3		DUT 4		DUT 5		DUT 6		DUT 7		DUT 8	
Bias polarity	> 0	< 0	> 0	< 0	> 0	< 0	> 0	< 0	> 0	< 0	> 0	< 0	> 0	< 0	> 0	< 0
R_{init}	3.9 k Ω		7.7 k Ω		20.2 k Ω		38.6 k Ω		122 k Ω		243 k Ω		322 k Ω		1.1 M Ω	
V_{min}	1.5	-1.6	1.5	-1.5	2.2	-1.8	1.1	-1.1	1.1	-1.1	1.6	-1.6	1.4	-1.3	0.9	-0.9
V_{max}	1.7	-1.8	1.9	-1.9	2.5	-2.1	1.5	-1.5	1.4	-1.4	2.0	-2.0	1.8	-1.7	1.2	-1.2
$C_0 (\times 10^{-2})$	1.6	5.1	4.1	-1.0	4.21	2.3	6.1	12	1.5	5.1	1.3	-1.1	3.2	-1.2	-0.87	-0.25
C_1	5.1	-3.8	24	-12	25	-41	19	-12	85	-54	796	475	4590	-1180	13422	-18704
$C_2 (\times 10^2)$	-1.2	-2.4	-3.4	0.76	-5.9	-14	-4.3	-5.9	-25	-3.3	-38	-29	-147	-39	123	247
$\alpha (\times 10^{-3})$	4.0		1.8		3.1		4.3		8.1		7.1		3.1		3.9	
β	-11		-5.6		-2.2		-7.7		-12.8		-91		-221		-871	

$$F(R, V) = C_0 R + C_1 V + C_2, \quad B(R, V) = \alpha R + \beta$$

Following the procedure described in sections II-B and II-D yields the corresponding surfaces for each of the programming (positive and negative) and read-out phases. The read-out data (all taken at 0.2 V but at *different* resistive levels) are used to extrapolate the baseline plane $B(R, V)$. The 0.2 V voltage was deemed to be non-invasive within the time frame of the experiment as has also been shown in the retention testing of [3] which uses the exact same stack. The baseline plane can now be calculated by extending a linear fit with slope α and intercept β among these points along the V axis

$$B(R, V) = \alpha R + \beta \quad (7)$$

As explained in section II we assume baseline to be independent from the applied voltage. Similarly, by fitting the plane

$$F(R, V) = C_0 R + C_1 V + C_2 \quad (8)$$

across all programming voltages will give us the *aggregate* uncertainty during programming. By subtracting this baseline from the fitted planes for positive and negative bias as per the equation (4) we can get the final switching surface $N(R, V)$. Surface $N(R, V)$ represents device noise that is purely attributed to switching rather than other external or internal factors. The same procedure has been followed for devices DUT 1 to DUT 8 that has been used in this paper to illustrate our modelling methodology. As the resistive is increasing from DUT 1 to DUT 8 it is apparent from the results of Table III that the overall noise floor is increasing as well (parameter β) which is consistent with the resistance broadening effect [10]. However switching noise is much less dependent on the resistive level itself once that noise floor has been removed (parameter C_0) which indicates that voltage (parameter C_1) is what primarily drives the uncertainty during programming for the particular technology used in the test case.

While the above analysis is a first estimate of switching noise for a memristive system it is important to mention some caveats. The described method does not deal with any hidden factors that can affect the switching of the device as is, for example, the curvature of the response of the device. Additionally our approach assumes that all distributions of data around a resistive state are Gaussian. Although for the devices used in this paper this is indeed the case, it is not

possible to say whether this is a universal behaviour among resistive devices or even if the resultant Gaussian distribution is a result of a series of variance propagations. An additional issue is that we are only using first-order approximation for the switching surfaces (i.e. planes). This might hide any additional structure present in the $\sigma(R, V)$ distribution. Additionally, we should mention that, our model does not deal with devices that are switching too rapidly as, for example, when the effect of bias vastly exceeds that of noise. In this particular case the consecutive data points of Figure 4a will lie on the vertical axis and the assumption of Gaussian distribution per triplet will no longer hold. This issue would also be the case for devices that exhibit strong bipolar behaviour expressed by abrupt switching between high and low resistive states as those, for example, shown in [20]. Finally, the overall philosophy behind our data-driven approach is that every individual device needs to be modelled. A full and more complete analysis of a *specific technology*, however, would require this procedure to be implemented on large numbers of devices and then followed by clustering and variation analysis on the parameters.

V. CONCLUSION

To summarise, in this paper we presented a way to model the switching noise of a memristive device while decoupling it from any underlying non-switching related forms of noise. Our approach starts by constructing the increment plots of each device ΔR vs. R and estimating the localised uncertainty on triplets of consecutive data points. Although this methodology exposes fine uncertainty patterns it does underestimate the actual switching noise. To counter this we introduced a statistical correction that is only dependent on the amount of neighbouring data points used for this estimation. By repeating the process separately for programming and read-out across many different programming voltages we can extract a $F(R, V)$ surface that models the uncertainty introduced during the switching procedure. The resulting surface is consisted of two separate planes, one for read-out and one for programming and provides a map of σ s that can be used to establish localised uncertainty distributions around a specific resistive state. By drawing a point from this distribution one can have a realistic noise approximation as an additional component existing switching behavioural models.

REFERENCES

- [1] L. Chua, "Memristor-the missing circuit element," *IEEE Transactions on Circuit Theory*, vol. 18, no. 5, p. 507–519, 1971.
- [2] D. B. Strukov, G. S. Snider, D. R. Stewart, and R. S. Williams, "The missing memristor found," *Nature*, vol. 453, no. 7191, p. 80–83, May 2008.
- [3] S. Stathopoulos, A. Khiat, M. Trapatseli, S. Cortese, A. Serb, I. Valov, and T. Prodromakis, "Multibit memory operation of metal-oxide bi-layer memristors," *Scientific Reports*, vol. 7, no. 1, Dec. 2017.
- [4] A. Serb, J. Bill, A. Khiat, R. Berdan, R. Legenstein, and T. Prodromakis, "Unsupervised learning in probabilistic neural networks with multi-state metal-oxide memristive synapses," *Nature Communications*, vol. 7, p. 12611, Sep. 2016.
- [5] J. Borghetti, G. S. Snider, P. J. Kuekes, J. J. Yang, D. R. Stewart, and R. S. Williams, "Memristive switches enable "stateful" logic operations via material implication," *Nature*, vol. 464, no. 7290, p. 873–876, Apr. 2010. [Online]. Available: <http://dx.doi.org/10.1038/nature08940>
- [6] G. A. Patterson, D. F. Grosz, and P. I. Fierens, "Noise on resistive switching: a fokker-planck approach," *Journal of Statistical Mechanics: Theory and Experiment*, vol. 2016, no. 5, p. 054043, May 2016.
- [7] P. Dutta and P. M. Horn, "Low-frequency fluctuations in solids: $1/f$ noise," *Reviews of Modern Physics*, vol. 53, no. 3, p. 497–516, Jul. 1981.
- [8] A. Stotland and M. Di Ventra, "Stochastic memory: Memory enhancement due to noise," *Physical Review E*, vol. 85, no. 1, Jan. 2012.
- [9] S. Choi, Y. Yang, and W. Lu, "Random telegraph noise and resistance switching analysis of oxide based resistive memory," *Nanoscale*, vol. 6, no. 1, p. 400–404, 2014.
- [10] S. Ambrogio, S. Balatti, V. McCaffrey, D. C. Wang, and D. Ielmini, "Noise-induced resistance broadening in resistive switching memory—Part II: Array statistics," *IEEE Transactions on Electron Devices*, vol. 62, no. 11, p. 3812–3819, Nov. 2015.
- [11] B. Gao, H. Wu, W. Wu, X. Wang, P. Yao, Y. Xi, W. Zhang, N. Deng, P. Huang, X. Liu, and et al., "Modeling disorder effect of the oxygen vacancy distribution in filamentary analog rram for neuromorphic computing," *2017 IEEE International Electron Devices Meeting (IEDM)*, Dec. 2017. [Online]. Available: <http://dx.doi.org/10.1109/IEDM.2017.8268326>
- [12] F. M. Puglisi, N. Zagni, L. Larcher, and P. Pavan, "Random telegraph noise in resistive random access memories: Compact modeling and advanced circuit design," *IEEE Transactions on Electron Devices*, vol. 65, no. 7, p. 2964–2972, Jul. 2018. [Online]. Available: <http://dx.doi.org/10.1109/TED.2018.2833208>
- [13] R. Degraeve, A. Fantini, N. Raghavan, L. Goux, S. Clima, Y. Y. Chen, A. Belmonte, S. Cosemans, B. Govoreanu, D. J. Wouters, and et al., "Hourglass concept for RRAM: A dynamic and statistical device model," *Proceedings of the 21th International Symposium on the Physical and Failure Analysis of Integrated Circuits (IPFA)*, pp. 245–249, Jun. 2014. [Online]. Available: <http://dx.doi.org/10.1109/IPFA.2014.6898205>
- [14] J. P. Strachan, A. C. Torrezan, F. Miao, M. D. Pickett, J. J. Yang, W. Yi, G. Medeiros-Ribeiro, and R. S. Williams, "State dynamics and modeling of tantalum oxide memristors," *IEEE Transactions on Electron Devices*, vol. 60, no. 7, p. 2194–2202, Jul. 2013. [Online]. Available: <http://dx.doi.org/10.1109/TED.2013.2264476>
- [15] S. Ambrogio, P. Narayanan, H. Tsai, R. M. Shelby, I. Boybat, C. di Nolfo, S. Sidler, M. Giordano, M. Bodini, N. C. P. Farinha, and et al., "Equivalent-accuracy accelerated neural-network training using analogue memory," *Nature*, vol. 558, no. 7708, p. 60–67, Jun. 2018. [Online]. Available: <http://dx.doi.org/10.1038/s41586-018-0180-5>
- [16] S. Menzel, U. Böttger, M. Wimmer, and M. Salinga, "Physics of the switching kinetics in resistive memories," *Advanced Functional Materials*, vol. 25, no. 40, p. 6306–6325, Jun. 2015. [Online]. Available: <http://dx.doi.org/10.1002/adfm.201500825>
- [17] I. Messaris, A. Serb, S. Stathopoulos, A. Khiat, S. Nikolaidis, and T. Prodromakis, "A Data-Driven verilog-a reram model," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, p. 3151–3162, 2018.
- [18] I. Messaris, S. Nikolaidis, A. Serb, S. Stathopoulos, I. Gupta, A. Khiat, and T. Prodromakis, "A TiO_2 ReRAM parameter extraction method," *2017 IEEE International Symposium on Circuits and Systems (ISCAS)*, May 2017.
- [19] R. Berdan, A. Serb, A. Khiat, A. Regoutz, C. Papavasiliou, and T. Prodromakis, "A uController-based system for interfacing selectorless RRAM crossbar arrays," *IEEE Transactions on Electron Devices*, vol. 62, no. 7, p. 2190–2196, Jul. 2015.
- [20] S. Brivio, E. Covi, A. Serb, T. Prodromakis, M. Fanciulli, and S. Spiga, "Gradual set dynamics in HfO_2 -based memristor driven by sub-threshold voltage pulses," *2015 International Conference on Memristive Systems (MEMRISYS)*, Nov. 2015. [Online]. Available: <http://dx.doi.org/10.1109/MEMRISYS.2015.7378383>