Designing a Bidirectional, Adaptive Neural Interface Incorporating Machine Learning Capabilities and Memristor-enhanced Hardware

Sergey Shchanikov[a,b*], Anton Zuev[a], Ilya Bordanov[a], Sergey Danilin[a], Vitaly Lukoyanov[b], Dmitry Korolev[b], Alexey Belov[b], Yana Pigareva[c], Arseny Gladkov[d], Alexey Pimashkin[c], Alexey Mikhaylov[b], Victor Kazantsev[c], Alexantrou Serb[e]

[a]Department of Information Technologies, Murom Institute Vladimir State University, Murom, Russia

[b]Research Institute of Physics and Technology, Lobachevsky University, Nizhny Novgorod, Russia

[c]Department of Neurotechnologies, Lobachevsky University, Nizhny Novgorod, Russia

[d]Neuroengineering Lab, Lobachevsky University, Nizhny Novgorod, Russia

[e]Department of Electronics and Computer Science (ECS), University of Southampton, Southampton, United Kingdom

***Corresponding author:**
Sergey Shchanikov
seach@inbox.ru

Keywords: memristors, artificial neural network, design, simulation, accuracy, neural interface, neuro-hybrid systems, neuromorphic computing

Abstract

Building bidirectional biointerfaces is one of the key challenges of modern engineering and medicine, with dramatic potential impact on bioprosthetics. Two of the major challenges of biointerface design concern signal stability and power efficiency. The former entails: a) ensuring that biosignal inputs corresponding to the same ground truth (e.g. patient "intentions") are recorded and interpreted consistently and b) maintaining the mapping from biointerface stimulation outputs to behavioral outputs (e.g. muscle movements). In this work we demonstrate how machine learning techniques, state-of-art nanoelectronics and microfluidics can combine forces to build and test low-power, adaptable biointerfaces that address both key challenges. Specifically, we demonstrate that: 1) we can emulate the input/output transfer characteristics of a structure biological neural network (BNN) with an artificial one (ANN), 2) it is possible to translate the resulting, "ideally trained" ANN into a hardware network using RRAM devices as synapses without significant loss of accuracy, despite concerns in the community about RRAM device reliability and 3) using a very simple mechanism of shifting the active stimulation electrode can fully restore functionality after the initial stimulation site degrades, prolonging the usable lifetime of the biointerface significantly. In this manner we place a key stepping stone towards building self-adjusting, low-power biointerfaces, themselves a foundational stepping stone towards adaptable, low-power bioprostheses.

1 Introduction

One of the most strategic areas of current research is in artificial bio-inspired systems, from general machine learning accelerator implementation to neuroprosthetics [1][2][3]. Amidst a wealth of proposed hardware implementations spanning from digital [4][5] to neuromorphic[6][7], approaches based on memristive (RRAM) technology [8][9][10] show considerable promise for increasing efficiency beyond the current state of art [11][12][13]. RRAM devices are thin-film structures [14][15] that change their resistance under suitably strong voltage or current bias [16][17]. Useful properties or RRAM include their low power and high speed operation [18], extreme scalability [19] and high memory density [20] amongst others.

In recent years, significant progress has been made in fabricating large RRAM crossbar arrays integrated with mixed signal peripheral circuits for hardware implementation of basic vector-matrix multiplication (VMM) operations [12], as well as more complex artificial neural network (ANN) circuits [21][22][23]. Furthermore, there has been work towards coupling RRAM-based systems with biological neurons in order to form neurohybrid systems [24], most recently demonstrated through a series of experiments where electrophysiological signals [25][26] and processed spiking events [27] were fed into RRAM devices for further refinement. In particular, in [27], spiking events from a biological culture and an artificial neuromorphic chip were exchanged over a RRAM, plastic synaptic array. This research direction is part of an ambitious drive towards intelligent, efficient, on-node neurointerfaces [1]. This can be contrasted to current approaches to building bidirectional, real-time neural interfaces, which currently require complex neuromorphic processors and/or spiking neural networks or specialized, software-implemented mathematical models [1][28][29][30].

However, RRAM-based systems also involve challenges, most notably RRAM device variability. As a result, the nominal quality of ANN operation achieved at the design stage is reduced in real operating conditions, potentially down to complete loss of functionality. Uncertainty factors include manufacturing variability [31], intrinsic behavioral variability of RRAM devices [32][33][34], operational parameter uncertainty [35], as well as higher level uncertainty arising through the architecture of ANNs [36][37]. Thus, a key research task is towards design and technological solutions towards the implementation of reliable RRAM-based ANNs (or "ANNs with memristors" - ANNM).

In this work we create and experimentally demonstrate a stable infrastructure base for the development of hybrid analog-digital neuroprocessors and systems based on memristors and connected to living neuronal cultures. The system combines for the first time: 1) a RRAM synaptic crossbar array, 2) an ordered cell culture substrate structure for forming a directed living network on a microfluidic setup and 3) an ANN module (in this case implemented as an ANNM) for automatically determining the optimal locations for stimulation within a multi-electrode cell culture set-up.

Overall, our work demonstrates: a) that all the pieces of core technology are in place for building bioprostheses with adaptive stimulation designed to counteract irreversible changes of the network structure and decline of interface efficacy, e.g. chronic signal degradation; one of the major failure points of modern bioprosthetics and b) a cohesive and complete test-bench for developing such systems ex-vivo; crucial because it allows experimentation with a solid degree of bio-realism. Our system constitutes a significant technical improvement over previously demonstrated systems [27]. The major differences vs. the previous state-of-art are: 1) increased network scale (from 2 to 160 artificial synapses), 2) introduction of a directed cell culture medium and 3) introduction of machine learning for auto-calibration of stimulation by use of our ANNM.

2 Overall Neural Interface Description

The bidirectional adaptive neural interface is designed to capture electrophysiological activity from living neuronal cultures and electrically stimulate them. The overarching functional structure of the system is illustrated in **Figure 1A**. The first one is the ANNM (Figure 1A, 1). It is trained to solve the nonlinear classification problem of finding stable characteristic patterns in the waveforms registered from the neuronal culture. ANNM operation is controlled by a bespoke software. Next, a TCP/IP-based software module connects the ANNM subsystem with the recording/stimulation subsystem (Figure 1A, 2), and finally, the physical recording and stimulation of the neuronal culture is carried out by a microelectrode array (MEA) module (Figure 1A, 3). The MEA module contains the actual electrode array together with its own software interface for capturing electrophysiological activity and directing stimulation. In this particular work, the MEA is connected to a neuronal culture hosted by a microfluidic chip, but in general it can be connected to any biological system (see neuroanimat [38] for example).

In this work, the neural interface is implemented concretely as shown in **Figure 1B**. The control centre of the system is an FPGA. This is connected to a PC which allows user interfacing (e.g. for controlling the operation flow). Next, the ANNM module consists of two parts: the artificial synapses and the artificial neurons. The synapses are implemented as 16x16 memristive crossbar arrays [39][40]. The neurons consist of electronic components which perform the main artificial neuron functions of accumulation and spike firing.

3 System components

3.1 Electronic set-up

3.1.1 Memristive devices

Memristive devices were fabricated on the basis of a newly engineered Au/Ta/ZrO_2(Y)/Ta_2O_5/TiN/Ti multilayer structure, which contains self-organized interface oxide layers, nanocrystals and is specially developed to obtain robust resistive switching with low variation of parameters [40][41]. An array of memristive devices is mounted into a standard metal-ceramic package (see **Figure 2A**) and can be easily integrated into the neurointerface circuit. Memristive devices demonstrate bipolar switching of anionic type between the high resistance state (HRS) and low resistance state (LRS). Both states are characterized by nonlinear current-voltage characteristics (CVC) (see **Figure 2B**) and low resistance variation extracted from CVC at voltage = 0.5V (see **Figure 2C**). It is worth noting that such nonlinear characteristics are appropriate for the formation of passive cross-bar arrays with state-of-art density [21].

To program the resistive states of our memristive devices, (which determine the corresponding synaptic weights in the ANNM) we used a variation of the incremental step pulse programming technique (ISPP) [42], adapted to the peculiarities and operating parameters of our memristive structure [15]. A feature of the developed technique is the use of active feedback when setting a specified state with the required accuracy, which reduces the total number of actions on memristive device in comparison with other techniques. In preparation for ANNM weight programming, a useful range of resistive states was identified through analysis of its DC CVCs. Next, programming was carried out by applying voltage pulses with increasing amplitude in the range from 0.5 to 3V (pulse duration of 1ms) with read-out after each pulse (at 0.1V, 1ms). In the event that the required resistance is not achieved within tolerance even after pulsing at the maximum acceptable amplitude, the memristive device is set to its initial state by applying an initializing voltage pulse (-3V, 1ms) with a current limit of 300μA, and the ISPP algorithm is repeated until the specified state is achieved. An example of a time series with two incremental step pulse ramp iterations the resulting read-outs illustrating behavior outside and inside a suitably defined valid range is shown in **Figure 2D**.

Furthermore, in **Figure 2E** we show additional results used to confirm the stability and multi-level capability of our devices. The test is carried out as follows: First, we set the devices to LRS using a -3V SET pulse

(reaching typ. ~6kΩ). Then we set the target resistance to ~10kΩ and begin the ISPP. If the target is achieved programming stops and a 5s retention under 1V stress commences. In that period, we take 160 resistance measurements (at 1V) to check for drift. Immediately afterwards, an additional 5s retention run is carried out under normal read-out conditions (0.1V). Once this cycle is complete the target resistance is moved to the next, immediately higher target (15kΩ) and the "programming-stress retention-no stress retention" cycle begins anew. This process continues until the final target value of 60kΩ. Once all target values have been swept, the full procedure is repeated; 20 times in total. At the end of the experiment, data measured under no stress retention is pooled across all 20 iteration and grouped by target resistance, yielding the histogram in **Figure 2E**. This proves the ability of our devices to assume multiple, unambiguous resistive states despite the combined effects of inherently imprecise programming and read-out uncertainty.

Throughout this work, memristors are programmed within ±15% tolerance for any state, as can be seen from **Figure 2E**. However, because of drift and electrical noise we increase the assumed 'working error margin' to ±20%. The histograms in **Figure 2E** and similar measurements confirm the sufficiency of this value. Estimated ANNM accuracy rates (see section 4) are based on this simplifying assumption. We also tested that each set of 160 no stress retention measurements follows a Gaussian distribution (Pearson test, p=0.05).

3.1.2 ANNM operating circuits

The typical artificial neuron consists of synapses, an accumulator and an activation function. In recent years there has been considerable interest in using memristors as synapses due to their high memory density[20], compactness[19] and fine, quasi-continuous tuneability[43].

Memristive devices can be arranged in arrays (discrete [22] or on-chip [9]), which may include active selectors [18] or be left selectorless [21] and operate either in individual regime (1 memristor = 1 memory/weight value) [44] or paired (2 memristors = 1 memory/weight) [18]. In this work we use on-chip, selectorless, paired arrays. The choice of a selectorless architecture was taken because the simplicity of the design overcomes the selectorless operation penalties for small arrays [21][45]. The choice of paired arrays reflects the easier implementation of bipolar weights using this configuration.

Our implementation is illustrated in **Figure 3** and operates as follows: During the write operation we apply a desired programming voltage V_{wr} (changing from the minimum value $V_{wr,min}$ to the maximum value $V_{wr,max}$) to the corresponding column and $V_{wr,max}/2$ (1.5V in our case) to all non-target columns [46]. Simultaneously, we apply 0V to the target row (connecting the target OpAmp's '+' terminal to the ground) and $V_{wr,max}/2$ to all other non-target rows (by applying the same voltage to V+ terminals). OpAmps U1 and U2 act as transimpedance amplifiers (TIAs) in this design. As a result: a) the target device sees $V=V_{wr}$ potential drop, b) all other devices on the same row see $V_{wr,max}/2$ voltage drop, c) all other devices on the same column see a maximum of $V_{wr,max}/2$ and d) all other devices see exactly 0V. Because we have tested that our memristive devices do not change their resistive state under 1.5V bias in any polarity, this scheme offers a good protection against cross-programming.

During the read operation we apply the read voltage V_{rd} (0.1V) to the corresponding column and 0V to all non-target columns. Simultaneously, we apply 0V to all rows (connecting OpAmps' '+' terminals to the ground). Then we read the voltage on the TIAs output (ADC1, ADC2) and calculate the current I_{TIA} through R_f. Thus, the conductance of a memristor is calculated as I_{TIA}/V_{rd}. In this manner, we implement the multiplication of input times weight via Ohm's law (with the weight encoded in conductance, not resistance) and the summation is carried out by Kirchhoff's current law. As a result, the output voltage of each OpAmp V_{SYN} under normal operation is given by:

$$V_{SYN} = -I_{IN} \cdot R_F = -\sum_k I_{IN,k} R_F \quad (1)$$

where I_{IN} is the total current from all active memristors k (whose input columns are set to the read voltage) and R_F is a value of feedback resistor.

Now, we observe that for each pair of memristors $x_{a,b}$ (e.g. M1 and M2 in **Figure 3**) $I_{IN,xa} = V_{in} \cdot g_{xa}$, which leads to:

$$V_{SYN} = -V_{IN,x} \cdot R_F \cdot g_{xa} \quad (2)$$

And therefore, the difference between the TIA output voltages for the pair becomes:

$$dV_{SYN} = -V_{IN,x} \cdot R_F \cdot (g_{xa} - g_{xb}) \quad (3)$$

allowing a natural, linear expression for the weight.

It is clear that for $g_{xa} > g_{xb}$ we encode positive weights and vice versa. This output voltage difference is captured by a standard difference amplifier (U_3) and amplified by an additional gain $K_{DIFF} = \frac{R_2}{R_1}$. This directly serves as the slope of our saturating linear activation function, which has been shown to be a good and computationally lightweight substitute for more traditional sigmoidal functions.

Finally, we use an inverting amplifier (U_4) to implement an activation function, whose slope is equal to gain of the amplifier. Output swing is adjusted using the resistive divider to match the output swing to the input swing of the next layer or to an ADC. By changing R3 or R4 we can adjust the swing to different output structures.

Naturally, some memristive devices may suffer from "stuck-at" faults despite repeated attempts to RESET them (see section 2.2). Similarly to [21] we counter this by checking our memristors for functionality post-fabrication. If one element in a pair becomes stuck, then an attempt is made to adjust its counterpart in order to produce a viable weight. If both fail, the stuck-at values are registered and introduced in the overarching software operating the array.

For biasing and reading we use data converters (ADCs and DACs) with bipolar output voltage ranges (dual rail supply). Switches (S_1, S_2, S_3, S_4) controlled by FPGA is used for commutation of bias voltages (-3V and 1.5V) obtained with resistive dividers with a shared element. Furthermore, we monitor both the final output of each neuron and the intermediate TIA outputs via ADCs for debugging purposes.

A project of ANNM is a set of structural and functional models, necessary and sufficient to solve the tasks assigned in terms of reference. During the engineering design of ANNM in compliance with international standards in the field of electronics design (IEC, ISO), the accuracy, fault tolerance, reliability and performance of operation in normal conditions and under the influence of destabilizing factors should be determined.

3.2 Biological set-up

3.2.1 Microfluidic device fabrication

To obtain the spatially ordered neuronal cultures during their growth on MEA, the microfluidic chips were fabricated via polydimethylsiloxane (PDMS) moulding techniques. Standard two-layer lithography was used for mould fabrication. The microfluidic chip consists of two chambers for cell cultivation and 8 microchannels providing unidirectional axon growth from Source chamber to Target chamber. Microchannels design was based on a sequence of 3 or 4 triangle segments that facilitated the directed axon growth (see **Figure 4A**).

The surfaces of the prepared PDMS chips were mounted with MEAs, which were coated with the adhesion-promoting polyethyleneimine molecules at the concentration of 1 mg/mL (Sigma-Aldrich, USA) and laminin at the concentration of 20 mg/mL (Sigma-Aldrich, USA). The chips were manually aligned with the MEA, which was composed of 60 electrodes (TiN electrodes, diameter 30 µm with 200 µm in between, Multichannel Systems, Germany), via a three-dimensional mechanical micromanipulator under a binocular. Furthermore, 14 or 24 electrodes were placed in each chamber, 24 or 32 electrodes were placed in the microchannels (3 or

4 electrodes in each of 8 microchannels) (see **Figure 4B**). We used reversible bonding for MEAs to prevent damage of the electrodes. After the PDMS chips were mounted to the MEA, it was cured in an oven at 80 °C for 30 min.

3.2.2 Cell culturing

Hippocampal cells were dissociated from embryonic mice (E18) and plated in the cell chambers of PDMS chips at an initial density of approximately 7,000–9,000 cells/mm^2. Mice were euthanized via cervical dislocation according to protocols approved by the Bioethics Committee of Lobachevsky University and carried out in accordance with Act708n (23.08.2010) of the Russian Federation National Ministry of Public Health, which states the rules of laboratory practice for the care and use of laboratory animals, and the Council Directive 2010/63 EU of the European Parliament (September 22, 2010) on the protection of animals used for scientific purposes. All efforts were made to minimize suffering. For culturing procedure details see [47]. The cells were cultured under constant conditions of 37 °C, 5% CO_2 in a humidified cell culture incubator (MCO-18AIC, SANYO, Japan).

3.2.3 Electrophysiology

Extracellular potential measurements were performed after 20 day *in vitro* when two cultures in the microfluidic device were already coupled by the axons through the microchannels and generated spontaneous activity. Detection of the recorded spikes was based on the threshold calculation of the signal median as described in our previous studies [47,48]. Signal analysis and statistics were performed with custom made software in MATLAB.

Stimulation through the MEA was performed with the STG-4004 pulse generator (Multichannel Systems, Germany). Series of 30 stimuli at 0.3 Hz were applied to each of four chosen electrodes out of 24 in the chamber A (**Figure 4A**). Low-frequency stimulation consisted of biphasic voltage pulses ±800 mV, 260 µs per phase, positive first. Signals from the electrodes placed in 4 microchannels were recorded by the MEA system (Multichannel Systems, Germany) at a sample rate of 20 kHz. An example of stimulus-evoked activity recorded from one of the electrodes (4th Line, 4th Row) is shown in **Figure 4C.** The post stimulus time histogram (PSTH) was calculated by taking all stimulus-evoked activities that follow each stimulus. The number of spikes occurring inside each 1-ms bin was counted and divided by the number of stimuli. Note the registered response consisted of direct axonal and synaptically evoked spikes which might be the result of a summation of the signals from various neuronal pathways. Example of the PSTH for the same electrode as in **Figure 4C** is shown in **Figure 4D.** Network responses to stimulation of two different electrodes of chamber A are shown in **Figure 4E**.

3.3 Machine learning set-up

The main task of the ANNM subsystem is to recognise spatio-temporal neural activity patterns that occur in response to stimuli applied to the neuronal culture.

Input data: In this study, we recorded the arrival times of the first four spikes registered in each of the four microfluidic channels of the chip in response to stimulation at one of four sites S_1, ..., S_4 within chamber A over a recording window of 50ms. The regularity of spike arrival times after stimuli from sites S_1, ..., S_4, allowed us to use said timings for classification. Each spike pattern was encoded into a vector of 16 spike arrival times (4 times-of-arrival (ToA) x 4 channels). These raw ToAs were then recognised and mapped to a given voltage amplitude range, each timing is divided by 50 and rounded to the specified bit depth of DACs. Bipolar 12-bit DACs with an output voltage range of ±5V were used. Thus, all values of the training and test sets are rounded off with a step of 10V/4096≈0.00244 V. Each stimulus site in chamber A is mathematically represented by a vector of four elements containing “1” at the position of current stimulus site S and “-1” at the remaining positions through the signal patterns on inputs. For spontaneous signals, a target vector consists of four “-1”. It is these vectors that we attempt to recover from ToA data.

Machine-learning network structure: We used an MLP architecture (see **Figure 5**). Due to the fact that the signals registered in microchannels are not orthogonal and can have high cross-correlation, the task is not linearly separable, so a double-layer architecture is necessary. The MLP receives 16 inputs corresponding to 4 microfluidic channels x spike ToAs over the 50ms window (about 3-4 spikes), then sends these to 8 hidden layer neurons (this is the maximum number of neurons our system supports on a single 16 × 16 memristor array). Finally, our 4 output neurons correspond to the four sites of stimulation S1, ..., S4 in chamber A. The activation function is saturating linear, which is closer to the natural output of differential amplifiers. Total number of synapses is 160 and memristors is 320. Biases are implemented in digital hardware for providing additional ability to tweak.

Overall machine-learning procedure (see **Figure 6**): First, we choose the 4 stimulation sites in chamber A in a manner that maximises the differences between the outputs produced by each site and register the outputs and behaviour of the ANNM. Then, we train an ANN off-line to correctly map the given input patterns to the known ground truth. RRAM devices that suffer from "stuck-at" faults are handled at this stage: they are discovered during ANNM operation and this information is fed into the software ANN in order to constrain its own training. The result is a proposed weight-set computed by the ANN. This is then programmed to the ANNM as accurately as possible. The ANNM then continues operating in open-loop, using input data to resolve which site received the stimulation. However, there is a further adjustment strategy: requests to test and if necessary adjust the stimulation configuration are raised under two possible conditions: 1) In response to performance droop: because we know the ground truth (what has been stimulated when) we can determine when the system begins to incorrectly return "no class recognized" as an output (e.g. due to natural signal drift as a result of the evolution of the biological network) too frequently – as determined by appropriately adjusted statistical tests. 2) Periodically as a background check: these occur regardless of running performance level. Once the system receives a stimulation test/adjust directive, it attempts to stimulate the target site (e.g. S1) by using different available electrodes in the vicinity of the electrode that was hitherto actually used for stimulating target S1 (radius 1 Moore neighborhood). All neighbors are probed, and the best candidate becomes the new preferred stimulation site for the corresponding target site (e.g. S1). After stimulus adjustment the ANNM may need to be trained anew if the initial degree of accuracy is too low. This way, we obtain continuously adaptive stimulation.

ANN to ANNM mapping procedures: Training consists of two steps: 1. Calculating synaptic weights off-line; 2. Programming synaptic weights on the device. A similar approach is described, for example, in papers [21,49] and makes it possible to provide the required accuracy of ANNM training. To calculate the weights, the ANNM control subsystem expects to receive a training sample from the system of registration and stimulation. After receiving it, the model calculates the weights and the corresponding conductivities of memristive devices. In this work, the calculation of weights was conducted using truncated Newton method [50]. The calculation of conductivities for each obtained weight value is performed using the equation (3). Weight programming is described in section 3.1.2. The performance of the ANNM during the training process is evaluated using the Mean Squared Error (*MSE*). The performance of the ANNM during the operation is evaluated using the Probability of Error (P_{err}), which indicates the number of correctly classified signal patterns compared to the total number of patterns. Additionally, in order to take the practical imperfections of physically implemented systems (e.g. uncertainty in programming RRAM devices [35]) we have used previously developed techniques for keeping practical errors within acceptable tolerance levels [51,52].

4 Results and Discussion

An experimental run was carried out as per the procedures outlined in section 3.3 (see flow chart in **Figure 6**). First, the neural cell culture was stimulated at four distinct sites, creating distinguishable output patterns. Next, a subset of the data (the training set) was fed into an ANN, which was trained to pattern match inputs to outputs. Subsequently, the ANN was transformed into an ANNM and the ANNM's performance was assessed. For this particular work every part of the experimental set-up was implemented physically, except the RRAM devices, which were represented by their statistics of measured achieved programming accuracies (hardware-in-the-loop methodology) described in section 3.1.1. This provides a reasonable estimate of performance ahead of the final phase (complete integration of the system).

The experimental results have allowed us to evaluate mean values and standard deviations of spike timings. Then we approximated the timing data with a Gaussian function and generate further, synthetic spike patterns pseudo-randomly for the four sources of stimuli (4000 synthetic signal patterns, 1000 for each stimulus) (see **Figure 7B**). This is necessary to increase the size of training and test sets for the developed ANNM. Any pattern where any of the spikes arrived with a time offset of more than evaluated, was classified as extraneous (S_r). On the basis of extraneous patterns, further 4000 pseudorandom sequences were synthesized as a control group. In total, 8000 spike activity patterns were mixed pseudorandomly and divided into a training set (743 for S_1, 740 for S_2, 747 for S_3, 744 for S_4, 3026 for S_r, total: 6000) and a test set (257 for S_1, 260 for S_2, 253 for S_3, 256 for S_4, 974 for S_r, total: 2000). The resulting training curve is shown in **Figure 8A**. Upon the end of the training phase (200 epochs) the obtained MSE value was 0.01324, translating to an ANN error probability of ~1.5% on the training set. On the test set the error is <2%.

Next, the ANN was mapped onto multiple instances of ANNMs. Because of the statistical nature of the ANN-to-ANNM mapping, each resulting ANNM was distinct and produced its own probability of error. Results based on 1000 ANNM instances yielded a maximum probability of error of 8.0%. The 99.5th percentile value was 6.8%, from which we conclude that error probability of <8.0% is routinely achievable if RRAM conductances can be controlled within our selected tolerance range of ±20% (which, in turn, we have shown to be routinely achievable in separate experiments on the RRAM array).

Next, we examine the influence of the adaptation/electrode adjustment process. Measured results are shown in **Figure 9**. After the initial set-up of the experiment we noted that stimulating site S_1 at electrode $E_{1,2}$ leads very reliably to the activation of the corresponding (correct) output neuron. The correct neuron always showed the highest degree of activation (0/20 hard errors), although on some occasions this activation was below the 0 threshold that we set for positive recognition of the input (2/20 soft errors). After letting the system run for 3 hours, we notice severe degradation of the signal, with 3/20 hard errors and 8/20 soft errors. The system then searches for better candidate electrodes for stimulation and eventually settles on electrode $E_{2,3}$. This results in a restoration of function with 1/20 hard and 2/20 soft errors. Importantly, this solid restoration of functionality is achieved even before the ANNM is retrained, showing the potential impact and usefulness of this methodology.

Finally, the results above indicate (proof-in-principle) that a complete, adaptable biointerface system, using ML techniques implemented on an ANNM and based on data from a directed microfluidic setup can provide practically usable results. The purpose of this work was to thoroughly explain and exhibit the multitude of components needed to achieve this and provide complete and accurate information on how these are operated and how they interact with each other. The next step is to piece everything together and demonstrate a full-hardware end-to-end experiment. Furthermore, we note a significant technological runway for our system, e.g. by incorporating additional cutting-edge neurophysiology techniques such as the "response-clamp" [53] which uses negative feedback to stabilise neural network responses.

Table 1: Key performance indicators measured from the machine-learning set-up. An idealized ANN achieves 1.5% error on the training set and 1.7% on the test set. When the ANN is translated to an ANNM with ±20%

accuracy of weight-to-RRAM conductance translation, we still obtain ~8% worst measured case test set error probability, with the 99.5th percentile error at 6.8%. Total number of ANNM instances: 1000.

ANN training set error probability	ANN test set error probability	ANNM (±20%) worst case test set error probability	ANNM (±20%) 99.5th percentile test set error probability
1.5%	1.7%	8.0%	6.8%

5 Conclusion

This work has illustrated, documented and demonstrated a set of key technologies that can be used to implement, test and continuously optimise the operation of a biointerface. Key results of the study include: A) That at least within the controlled medium of a directed cell culture biosignal transmission from selected stimulation sites to selected output sites is stable enough to admit emulation via an ANN. Moreover, the self-adaptability of the ANN in combination with perturbation analysis can provide a self-adapting stimulus protocol to mitigate natural drift in biosignal transmission properties within the culture. B) That ANNMs are credible candidates for implementing synapses in hardware: even with ±20% programming tolerance it is possible to obtain acceptably low error rate increases vs. the ideally trained ANN from which they were derived. This is critical because programming accuracies of well below ±20% have been experimentally demonstrated both by our consortium and others [43]. This also bodes well for robustness against natural weight drift over time. Together with continuous developments in RRAM technology [12] our work highlights the promise of this technology for delivering the low power biointerfaces of the future (for a review of RRAM technology benefits in general see [13][2]).

Long term, we consider this work to be a very useful stepping stone towards automated bioprostheses. We envisage futuristic systems, equipped with neural networks that effectively do what the brain does on a continuous basis: refine their ability to map input neural signals to desired behavioural outcomes (for example mapping signals travelling down a severed spinal cord to movements of limbs, either biological limbs stimulated directly at the neuro-muscular junction site or even entirely artificial limbs – the self-adaptation mechanism can work equally well for both cases). Moreover, we envisage all this capability being delivered at ultra-low power, courtesy of RRAM technology; delivering long battery lifetimes, or possibly such low power envelopes that the entire system moves within the realm of power harvesters. With this work we share with the community our progress so far in that direction.

6 Author Contributions

All authors gave substantial contribution to the development of this work equally, drafting and revising it critically; furthermore, all Authors approved its final version for publication.

7 Funding

The research was supported by the Russian Science Foundation (grant No. 16-19-00144).

8 Competing interests

The authors declare no competing interests.

9 References

[1] Vassanelli S, Mahmud M. Trends and challenges in neuroengineering: Toward “intelligent”

neuroprostheses through brain-"brain inspired systems" communication. Front Neurosci 2016;10. https://doi.org/10.3389/fnins.2016.00438.

[2] Mikhaylov A, Pimashkin A, Pigareva Y, Gerasimova S, Gryaznov E, Shchanikov S, et al. Neurohybrid Memristive CMOS-Integrated Systems for Biosensors and Neuroprosthetics. Front Neurosci 2020;14:358. https://doi.org/10.3389/fnins.2020.00358.

[3] Mehonic A, Sebastian A, Rajendran B, Simeone O, Vasilaki E, Kenyon AJ. Memristors—From In-Memory Computing, Deep Learning Acceleration, and Spiking Neural Networks to the Future of Neuromorphic and Bio-Inspired Computing. Adv Intell Syst 2020. https://doi.org/10.1002/aisy.202000085.

[4] Lines A, Joshi P, Liu R, McCoy S, Tse J, Weng YH, et al. Loihi asynchronous neuromorphic research chip. Proc. - Int. Symp. Asynchronous Circuits Syst., 2018. https://doi.org/10.1109/ASYNC.2018.00018.

[5] Debole M V., Taba B, Amir A, Akopyan F, Andreopoulos A, Risk WP, et al. TrueNorth: Accelerating From Zero to 64 Million Neurons in 10 Years. Computer (Long Beach Calif) 2019. https://doi.org/10.1109/MC.2019.2903009.

[6] Douglas R, Mahowald M, Mead C. Neuromorphic analogue VLSI. Annu Rev Neurosci 1995. https://doi.org/10.1146/annurev.ne.18.030195.001351.

[7] Ielmini D, Pedretti G. Device and Circuit Architectures for In-Memory Computing. Adv Intell Syst 2020. https://doi.org/10.1002/aisy.202000040.

[8] Waser R, Aono M. Nanoionics-based resistive switching memories. Nat Mater 2007. https://doi.org/10.1038/nmat2023.

[9] Xia Q, Yang JJ. Memristive crossbar arrays for brain-inspired computing. Nat Mater 2019;18:309–23. https://doi.org/10.1038/s41563-019-0291-x.

[10] Erokhin V. Memristive Devices for Neuromorphic Applications: Comparative Analysis. Bionanoscience 2020. https://doi.org/10.1007/s12668-020-00795-1.

[11] Qin Y-F, Bao H, Wang F, Chen J, Li Y, Miao X-S. Recent Progress on Memristive Convolutional Neural Networks for Edge Intelligence. Adv Intell Syst 2020. https://doi.org/10.1002/aisy.202000114.

[12] Amirsoleimani A, Alibart F, Yon V, Xu J, Pazhouhandeh MR, Ecoffey S, et al. In-Memory Vector-Matrix Multiplication in Monolithic Complementary Metal–Oxide–Semiconductor-Memristor Integrated Circuits: Design Choices, Challenges, and Perspectives. Adv Intell Syst 2020. https://doi.org/10.1002/aisy.202000115.

[13] Zidan MA, Strachan JP, Lu WD. The future of electronics based on memristive systems. Nat Electron 2018;1:22–9. https://doi.org/10.1038/s41928-017-0006-8.

[14] Ismail M, Abbas H, Choi C, Kim S. Controllable analog resistive switching and synaptic characteristics in ZrO2/ZTO bilayer memristive device for neuromorphic systems. Appl Surf Sci 2020;529:147107. https://doi.org/https://doi.org/10.1016/j.apsusc.2020.147107.

[15] Emelyanov A V., Nikiruy KE, Demin VA, Rylkov V V., Belov AI, Korolev DS, et al. Yttria-stabilized zirconia cross-point memristive devices for neuromorphic applications. Microelectron Eng 2019;215. https://doi.org/10.1016/j.mee.2019.110988.

[16] Chua LO. Memristor—The Missing Circuit Element. IEEE Trans Circuit Theory 1971;18:507–19. https://doi.org/10.1109/TCT.1971.1083337.

[17] Strukov DB, Snider GS, Stewart DR, Williams RS. The missing memristor found. Nature 2008;453:80–3. https://doi.org/10.1038/nature06932.

[18] Yao P, Wu H, Gao B, Tang J, Zhang Q, Zhang W, et al. Fully hardware-implemented memristor convolutional neural network. Nature 2020. https://doi.org/10.1038/s41586-020-1942-4.

[19] Pi S, Li C, Jiang H, Xia W, Xin H, Yang JJ, et al. Memristor crossbar arrays with 6-nm half-pitch and 2-nm critical dimension. Nat Nanotechnol 2019;14:35–9. https://doi.org/10.1038/s41565-018-0302-0.

[20] Kataeva I, Ohtsuka S, Nili H, Kim H, Isobe Y, Yako K, et al. Towards the development of analog neuromorphic chip prototype with 2.4M integrated memristors. Proc. - IEEE Int. Symp. Circuits Syst., vol. 2019- May, Institute of Electrical and Electronics Engineers Inc.; 2019. https://doi.org/10.1109/ISCAS.2019.8702125.

[21] Bayat FM, Prezioso M, Chakrabarti B, Nili H, Kataeva I, Strukov D. Implementation of multilayer perceptron network with highly uniform passive memristive crossbar circuits. Nat Commun 2018;9. https://doi.org/10.1038/s41467-018-04482-4.

[22] Mikhaylov AN, Morozov OA, Ovchinnikov PE, Antonov IN, Belov AI, Korolev DS, et al. One-Board Design and Simulation of Double-Layer Perceptron Based on Metal-Oxide Memristive Nanostructures. IEEE Trans Emerg Top Comput Intell 2018;2:371–9. https://doi.org/10.1109/tetci.2018.2829922.

[23] Prudnikov N V, Lapkin DA, Emelyanov A V, Minnekhanov AA, Malakhova YN, Chvalun SN, et al. Associative STDP-like learning of neuromorphic circuits based on polyaniline memristive microdevices. J Phys D Appl Phys 2020;53:414001. https://doi.org/10.1088/1361-6463/ab9262.

[24] Chiolerio A, Chiappalone M, Ariano P, Bocchini S. Coupling resistive switching devices with neurons: State of the art and perspectives. Front Neurosci 2017;11. https://doi.org/10.3389/fnins.2017.00070.

[25] Gupta I, Serb A, Khiat A, Zeitler R, Vassanelli S, Prodromakis T. Real-time encoding and compression of neuronal spikes by metal-oxide memristors. Nat Commun 2016;7. https://doi.org/10.1038/ncomms12805.

[26] Gupta I, Serb A, Khiat A, Zeitler R, Vassanelli S, Prodromakis T. Sub 100 nW volatile nano-metal-oxide memristor as synaptic-like encoder of neuronal spikes. IEEE Trans Biomed Circuits Syst 2018;12:351–9. https://doi.org/10.1109/TBCAS.2018.2797939.

[27] Serb A, Corna A, George R, Khiat A, Rocchi F, Reato M, et al. Memristive synapses connect brain and silicon spiking neurons. Sci Rep 2020;10. https://doi.org/10.1038/s41598-020-58831-9.

[28] Boi F, Moraitis T, Feo V De, Diotalevi F, Bartolozzi C, Indiveri G, et al. A bidirectional brain-machine interface featuring a neuromorphic hardware decoder. Front Neurosci 2016;10. https://doi.org/10.3389/fnins.2016.00563.

[29] Hogri R, Bamford SA, Taub AH, Magal A, Giudice P Del, Mintz M. A neuro-inspired model-based closed-loop neuroprosthesis for the substitution of a cerebellar learning function in anesthetized rats. Sci Rep 2015;5. https://doi.org/10.1038/srep08451.

[30] Buccelli S, Bornat Y, Colombi I, Ambroise M, Martines L, Pasquale V, et al. A Neuromorphic

Prosthesis to Restore Communication in Neuronal Networks. IScience 2019;19:402–14. https://doi.org/10.1016/j.isci.2019.07.046.

[31] Zhao M, Gao B, Tang J, Qian H, Wu H. Reliability of analog resistive switching memory for neuromorphic computing. Appl Phys Rev 2020. https://doi.org/10.1063/1.5124915.

[32] Stathopoulos S, Serb A, Khiat A, Ogorzalek M, Prodromakis T. A Memristive Switching Uncertainty Model. IEEE Trans Electron Devices 2019. https://doi.org/10.1109/TED.2019.2918102.

[33] Spagnolo B, Guarcello C, Magazzù L, Carollo A, Adorno DP, Valenti D. Nonlinear relaxation phenomena in metastable condensed matter systems. Entropy 2017. https://doi.org/10.3390/e19010020.

[34] Spagnolo B, Valenti D, Guarcello C, Carollo A, Persano Adorno D, Spezia S, et al. Noise-induced effects in nonlinear relaxation of condensed matter systems. Chaos, Solitons & Fractals 2015;81:412–24. https://doi.org/https://doi.org/10.1016/j.chaos.2015.07.023.

[35] Mehonic A, Joksas D, Ng WH, Buckwell M, Kenyon AJ. Simulation of Inference Accuracy Using Realistic RRAM Devices. Front Neurosci 2019;13. https://doi.org/10.3389/fnins.2019.00593.

[36] Torres-Huitzil C, Girau B. Fault and Error Tolerance in Neural Networks: A Review. IEEE Access 2017;5:17322–41. https://doi.org/10.1109/ACCESS.2017.2742698.

[37] Yeung D, Cloete I, Shi D, Ng W. Sensitivity Analysis for Neural Networks. Springer Berlin; 2010.

[38] Xydas D, Norcott DJ, Warwick K, Whalley BJ, Nasuto SJ, Becerra VM, et al. Architecture for neuronal cell control of a mobile robot. Springer Tracts Adv Robot 2008. https://doi.org/10.1007/978-3-540-78317-6_3.

[39] Gryaznov YG, Antonov I, Kotina A, Kotomina V, Mikhailov A, Sharapov A, et al. Russian sertificate of state registration of integrated circuit topology No. 018630129. Topology of test chip with array of memristive microdevices No. 2018630123. 2018630129, 2018.

[40] Mikhaylov A, Belov A, Korolev D, Antonov I, Kotomina V, Kotina A, et al. Multilayer Metal-Oxide Memristive Device with Stabilized Resistive Switching. Adv Mater Technol 2020. https://doi.org/10.1002/admt.201900607.

[41] Tihov S, Belov A, Korolev D, Antonov A, Sushkov D, Pavlov D, et al. Electrophysical characteristics of multilayer memristive nanostructures based on yttrium stabilized zirconium dioxide and tantalum oxide. Tech Phys 2020;90:298–304.

[42] Suh KD, Suh BH, Lim YH, Kim JRJK, Choi YJ, Koh YN, et al. A 3.3 V 32 Mb NAND Flash Memory with Incremental Step Pulse Programming Scheme. IEEE J Solid-State Circuits 1995. https://doi.org/10.1109/4.475701.

[43] Stathopoulos S, Khiat A, Trapatseli M, Cortese S, Serb A, Valov I, et al. Multibit memory operation of metal-oxide Bi-layer memristors. Sci Rep 2017. https://doi.org/10.1038/s41598-017-17785-1.

[44] Emelyanov A V., Nikiruy KE, Serenko A V., Sitnikov A V., Presnyakov MY, Rybka RB, et al. Self-adaptive STDP-based learning of a spiking neuron with nanocomposite memristive weights. Nanotechnology 2020;31:045201. https://doi.org/10.1088/1361-6528/ab4a6d.

[45] Hamdioui S, Aziza H, Sirakoulis GC. Memristor based memories: Technology, design and test. Proc. - 2014 9th IEEE Int. Conf. Des. Technol. Integr. Syst. Nanoscale Era, DTIS 2014, IEEE Computer

Society; 2014. https://doi.org/10.1109/DTIS.2014.6850647.

[46] Shevgoor M, Muralimanohar N, Balasubramonian R, Jeon Y. Improving memristor memory with sneak current sharing. Proc. 33rd IEEE Int. Conf. Comput. Des. ICCD 2015, 2015. https://doi.org/10.1109/ICCD.2015.7357164.

[47] Pimashkin A, Gladkov A, Mukhina I, Kazantsev V. Adaptive enhancement of learning protocol in hippocampal cultured networks grown on multielectrode arrays. Front Neural Circuits 2013. https://doi.org/10.3389/fncir.2013.00087.

[48] Pimashkin A, Kastalskiy I, Simonov A, Koryagina E, Mukhina I, Kazantsev V. Spiking signatures of spontaneous activity bursts in hippocampal cultures. Front Comput Neurosci 2011;5. https://doi.org/10.3389/fncom.2011.00046.

[49] Liu Z, Tang J, Gao B, Yao P, Li X, Liu D, et al. Neural signal analysis with memristor arrays towards high-efficiency brain–machine interfaces. Nat Commun 2020. https://doi.org/10.1038/s41467-020-18105-4.

[50] Robinson SM. Numerical Optimization 2nd Edition. 2018. https://doi.org/10.1007/978-0-387-40065-5.

[51] Danilin SN, Shchanikov SA, Galushkin AI. The research of memristor-based neural network components operation accuracy in control and communication systems. 2015 Int. Sib. Conf. Control Commun. SIBCON 2015 - Proc., Institute of Electrical and Electronics Engineers Inc.; 2015. https://doi.org/10.1109/SIBCON.2015.7147034.

[52] Danilin SN, Shchanikov SA, Panteleev SV. Determining Operation Tolerances of Memristor-Based Artificial Neural Networks. 2016 Int. Conf. Eng. Telecommun., IEEE; 2016, p. 34–8. https://doi.org/10.1109/EnT.2016.016.

[53] Keren H, Marom S. Controlling neural network responsiveness: Tradeoffs and constraints. Front Neuroeng 2014. https://doi.org/10.3389/fneng.2014.00011.

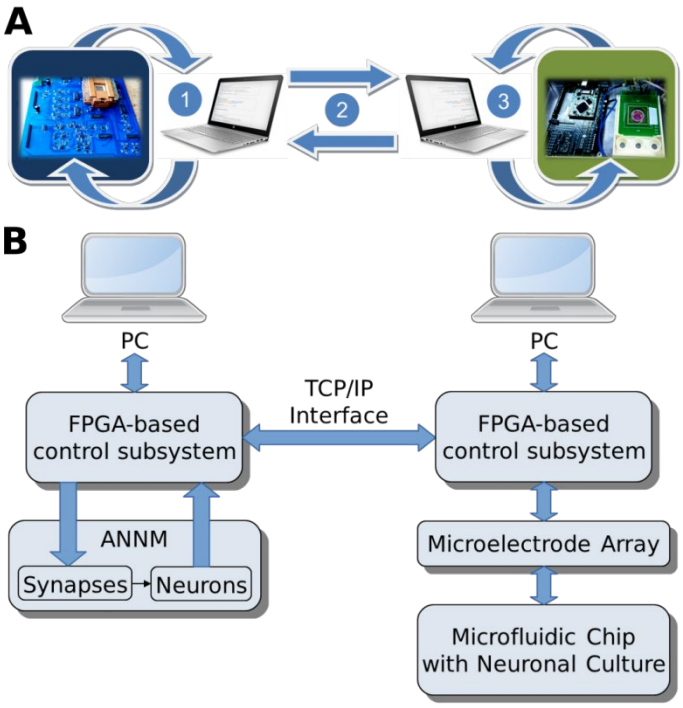

Figure 1 | Bidirectional adaptive neural interface architecture. **(A)** Functional structure, consisting of the following parts: (1) ANN set-up, (2) communications infrastructure, (3) MEA set-up. **(B)** Concrete implementation block diagram. The ANN and MEA sides both have FPGA-based control systems at their centres. These are operated by corresponding PC stations and orchestrate the activity of the ANNM and MEA components below them (in the MEA case including the microfluidic chip).

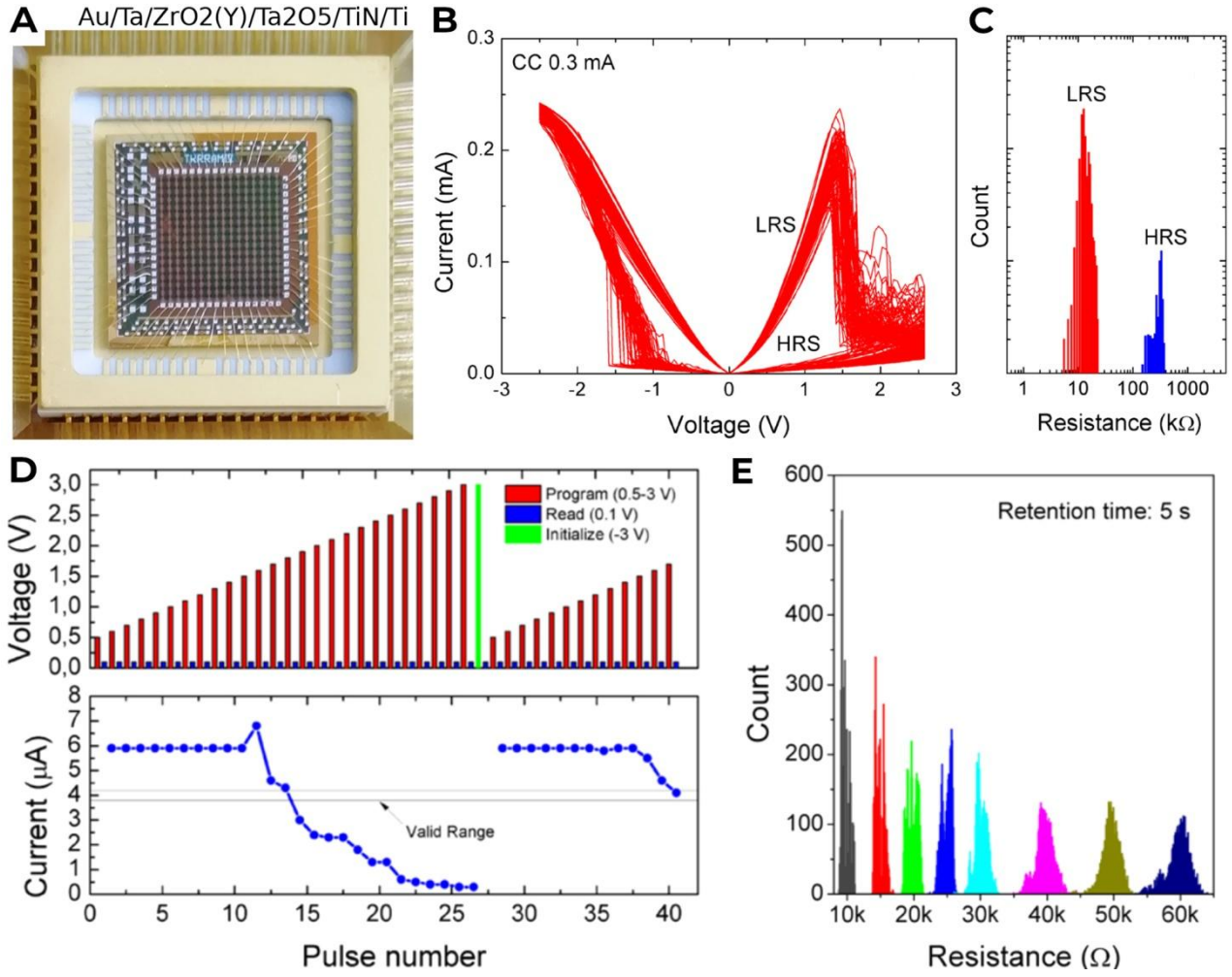

Figure 2 | Memristive microdevices. **(A)** Photograph of a packaged cross-bar array of memristive microdevices, **(B)** typical CVC in bipolar switching regime (100 cycles) and **(C)** distributions of resistive states over 1,000 CVC cycles extracted at 0.5V. **(D)** Voltage ramps applied to memristor and read-out current response during ISPP algorithm application. **(E)** Distributions of measurements taken at 8 different programmed resistive states showing that our devices support 8-state programming unambiguously in the presence of both programming and read-out uncertainties (see text).

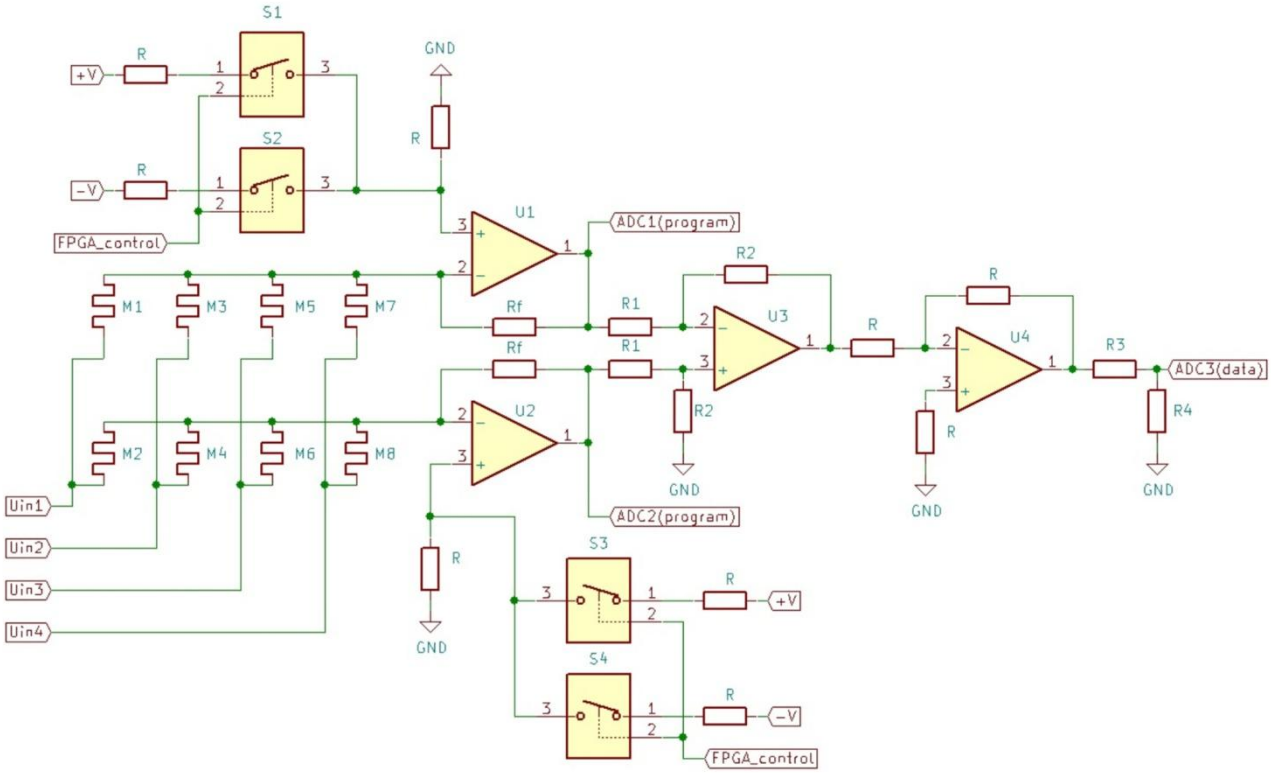

Figure 3 | Hardware implementation of the ANNM. A simplified circuit diagram of a neuron. Memristors-based synapses store weights and multiply input information by them. Input information at the physical level is voltage amplitude of an input signal. Multiplication is performed by Ohm's law. Addition is performed by Kirchhoff's rule. All synapses located in two adjacent lines belong to one neuron in order to obtain the bipolar values of weights. Neurons consist of differential amplifiers performed transfer functions.

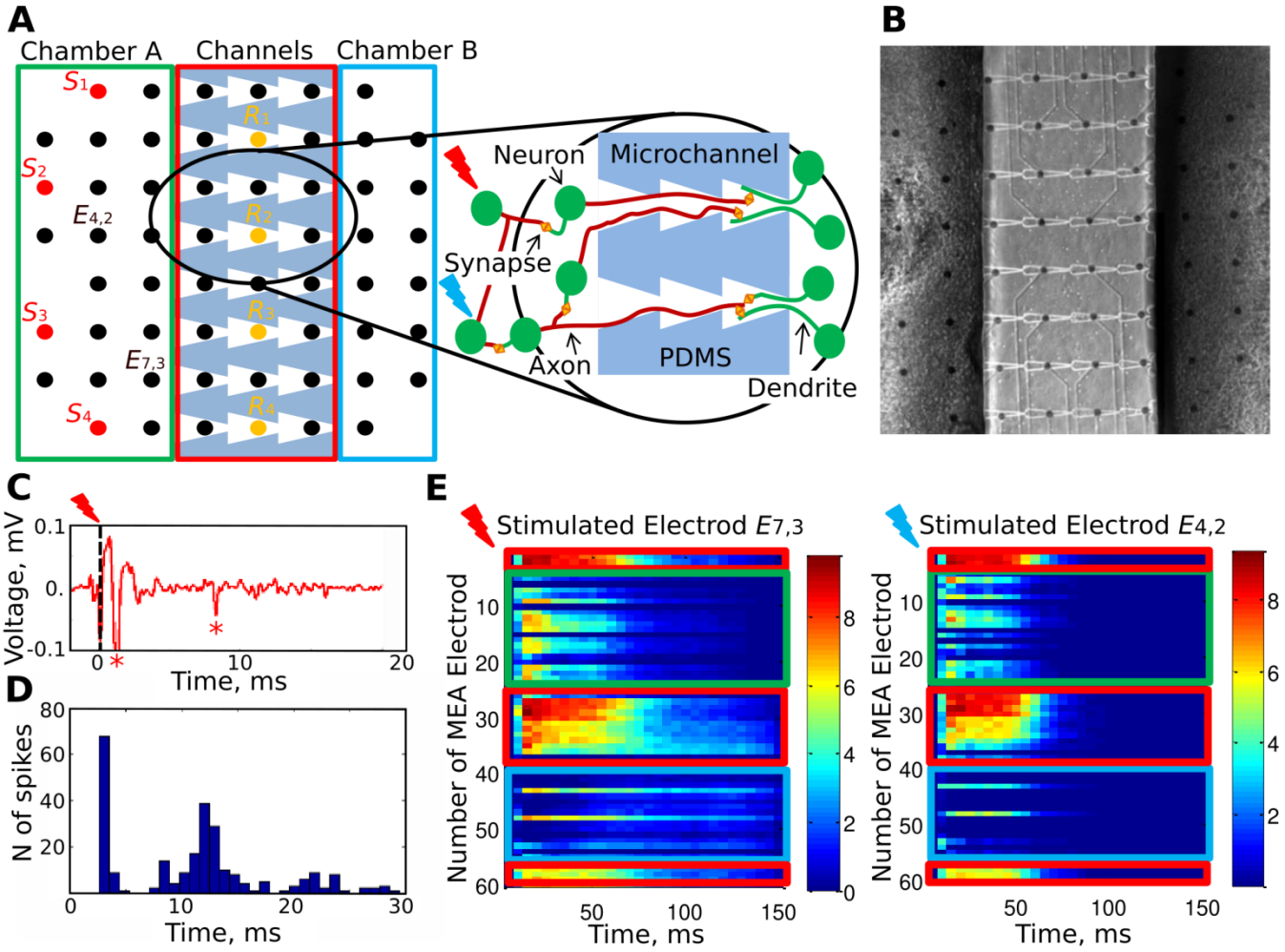

Figure 4 | Microfluidic chip (MFC). **(A)** Schematic illustration of neural network grown in microfluidic chip with ordered connectivity between two subnetworks. Green and blue boxes – chambers, red box – channels. The stimuli were applied to the electrodes colored by red (S_1, …, S_4), the neural activity was registered by electrodes colored by yellow (R_1, …, R_4). The input signal from the Source chamber is converted by network and integrated into the microchannels through the axons. **(B)** Neuronal cells plated in two chambers of a microfluidic chip and connected through microchannels for unidirectional axon growth (DIV 15) and combined with MEA, scale bar = 200 um. **(C)** Example of response recorded on electrode in the microchannel (4th Line, 4th Row) for single applied stimulus (see text): direct axonal (<5 ms) and synaptically evoked (8 ms) spikes are marked by asterisks. **(D)** Post-stimulus time histogram (PSTH) in response to 30 consecutive stimuli applied to the same electrode (see methods). First sharp peak is associated with direct axonal spike, second – with synaptically evoked responses with ~5ms jitter. **(E)** Examples of averaged network response registered from all 60 electrodes of the MEA to stimulation of two different sites (3rd Line, 7th Row and 2nd Line, 4th Row). Colour outlines correspond to chamber A, channel and camber B electrode groups as shown in (A). Colourbar: average number of spikes for each electrode per each 5ms time bin. Outline box colors correspond to (A).

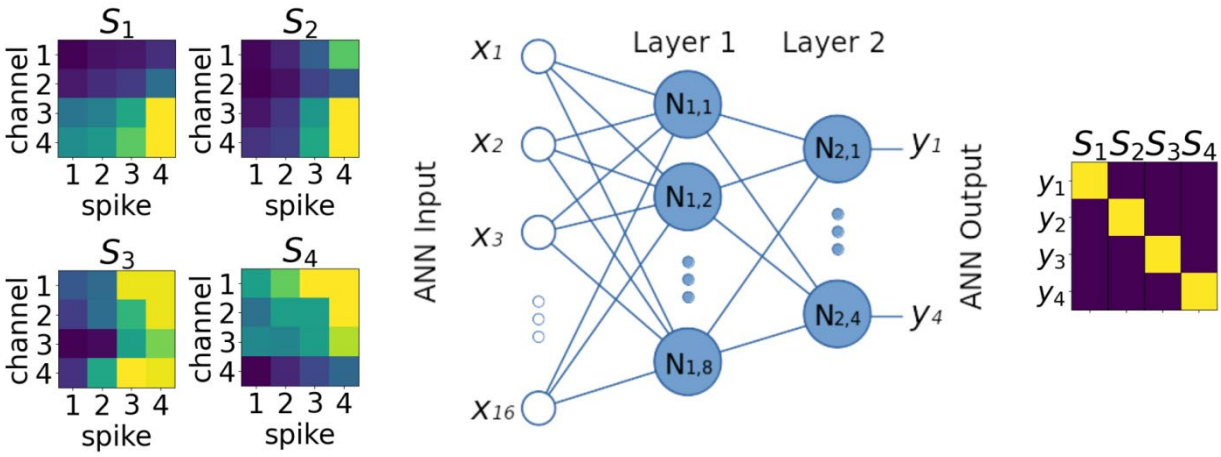

Figure 5 | ANNM architecture: We used double-layer perceptron network with 16 inputs, 8 neurons in the hidden layer, and 4 outputs. Each input sample (left side) is a spatio-temporal neural activity pattern, consisting of the arrival times of the first four spikes registered in the four microfluidic channels of the MFC in response to stimulation at one of four sites S_1, ..., S_4 in chamber A. Before processing, input patterns are transformed from a two-dimensional array to a one-dimensional array (of 4 * 4 = 16 arrival times). The perceptron classifies patterns by the physical location of living neuronal culture stimulation, which is indicated by the output signal (right side).

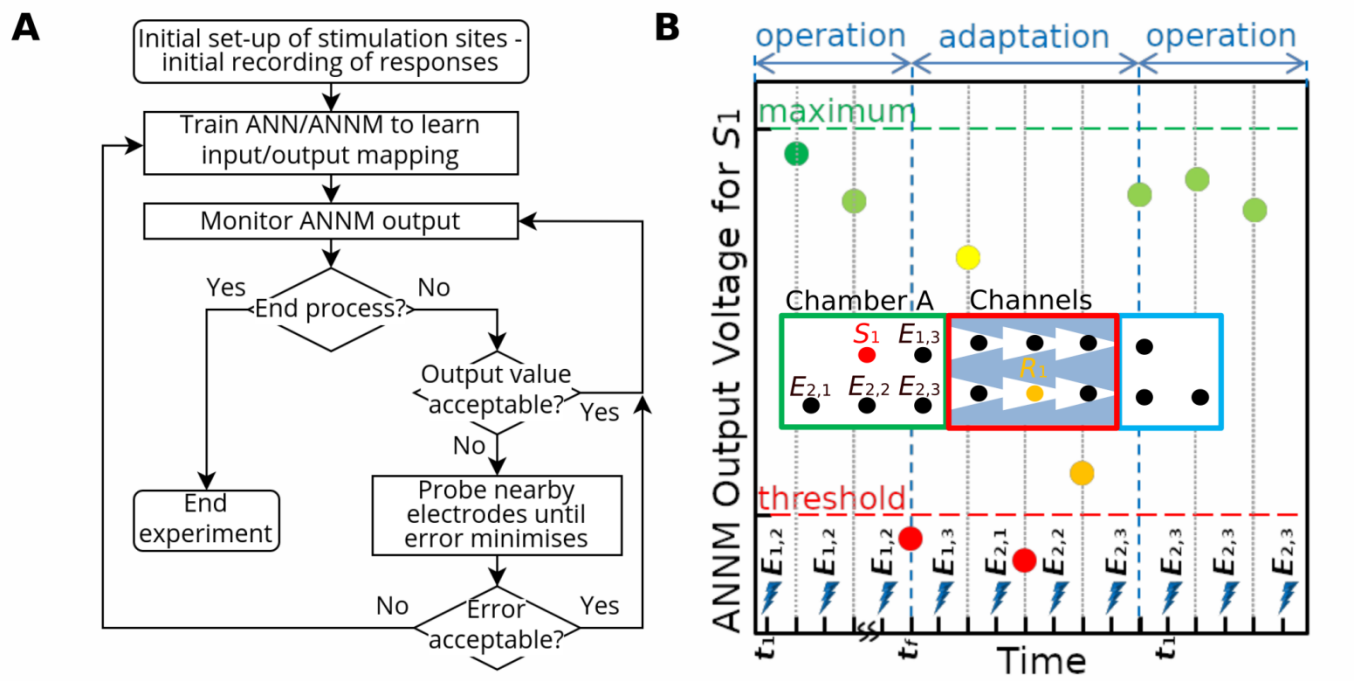

Figure 6 | Adaptation mechanisms in ANNM system. (A) Flow chart of adaptation strategy. Please note there are two processes of system adaptation: i) probing for a better stimulation site, ii) training the network at the chosen stimulation site. (B) Concept diagram of ANNM adaptation procedure. Assume that initially site S_1 corresponds to electrode $E_{1,2}$. Immediately after training the ANNM, we notice that the network responds very clearly to stimulation on $E_{1,2}$ by raising the "S_1" flag (t_1, t_2). As we continue stimulating, the signal begins to degrade with subsequent stimuli (t_3, t_4), until it drops below a minimum acceptable threshold (t_f). At that point the system probes nearby electrodes for a better stimulation site (t_{f+1}, …) until it finds the best candidate (t_{f+8}). In this case, $E_{2,3}$ exhibits a very good degree of accuracy and no further training is required (though this could in principle be done for further optimisation). Had $E_{2,3}$ still been the best candidate, but with accuracy below threshold, the ANNM would be explicitly trained on $E_{2,3}$ as per (A). Adaptation operations may be also triggered on-demand as a failsafe.

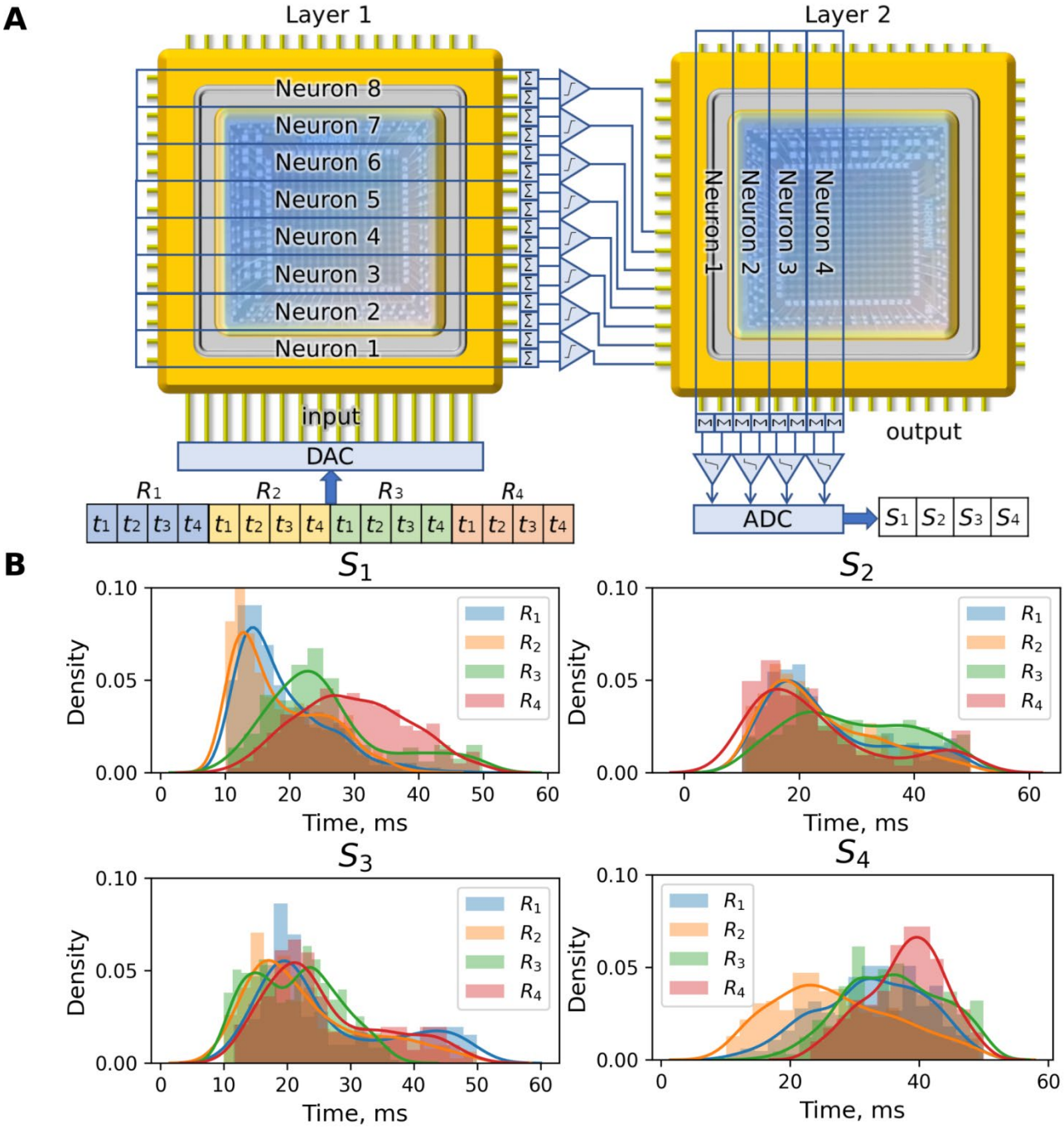

Figure 7 | Training and test sets. **(A)** Block diagram of the ANNM under development. (t_1, …, t_4): spike times-of-arrival (ToA) for first four spikes in each channel as indicated (color-coded to match panel **B**). (S_1, …, S_4): output vector indicating prediction of which site delivered stimulus. **(B)** Spikes registered in four channels in response to stimulation at sites S_1, …, S_4. Histograms of spike timings (t_1, …, t_4) are experimentally measured. In some channels, the 4th or sometimes even 3rd spikes occurred after >50ms, and were thus recorded as '50' for ANNM training purposes (corresponding to "1" after normalization).

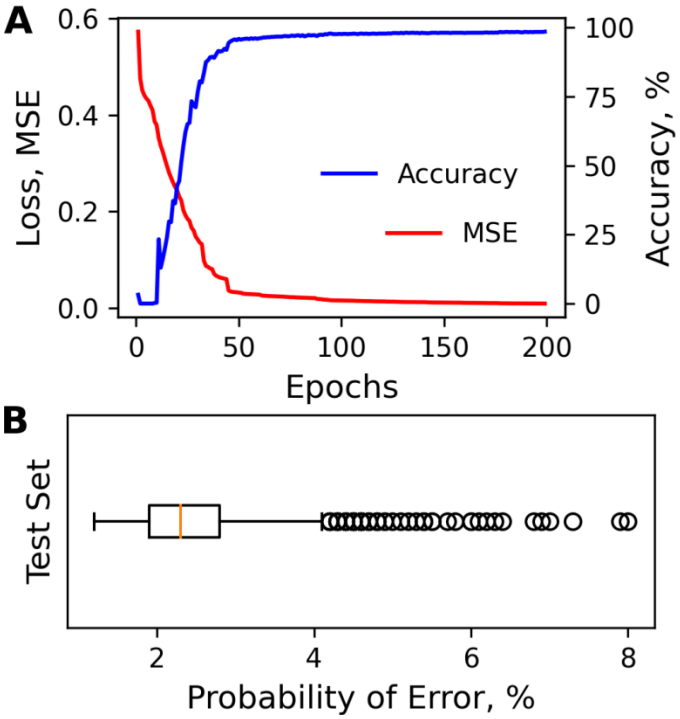

Figure 8 | Learning curve and error tolerance of ANN: **(A)** Learning curve for the MLP ANN on training set illustrating fast and accurate learning. 200 epochs total. Final MSE: ~0.009 for the training set and ~0.013 for the test set. **(B)** Results of mapping idealized ANN to ANNM using our ±20% 'working uncertainty' value in setting the ANNM weights. 1000 instances of ANNMs. Max. error rate: 8.0%. 99.5th percentile: 6.8%.

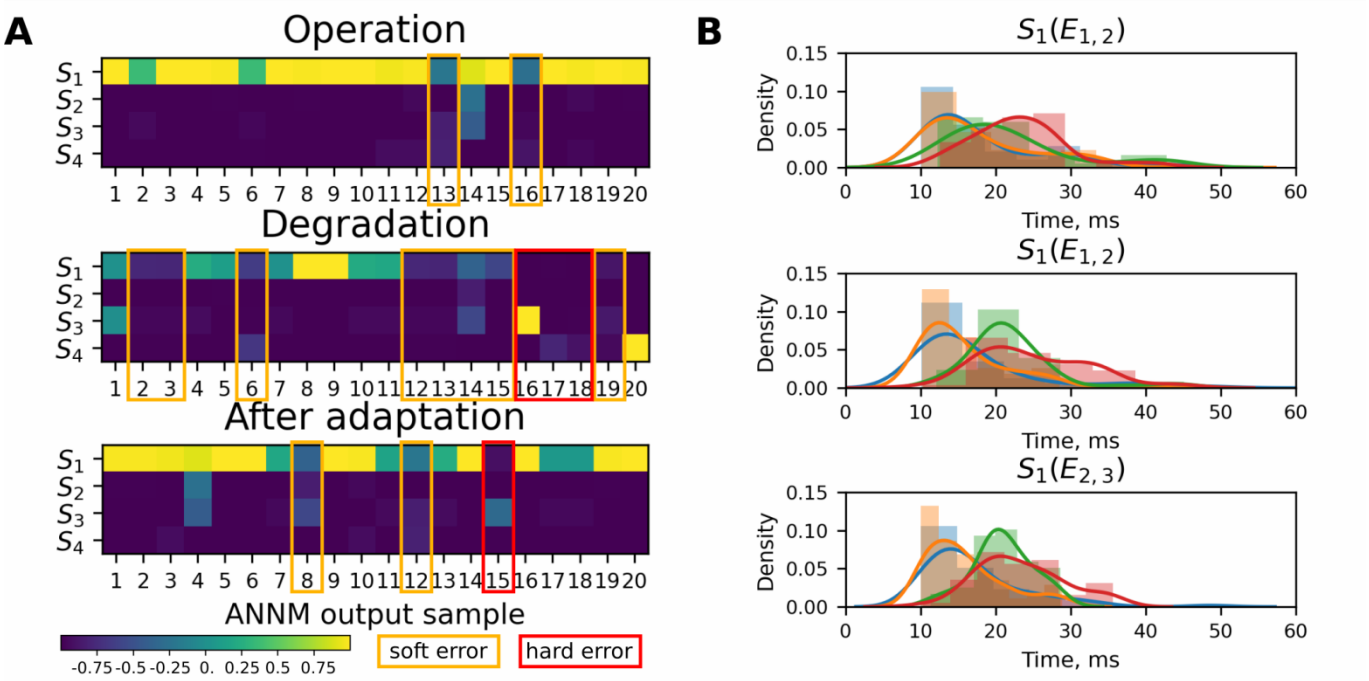

Figure 9 | (A) Example of electrode adaptation process for a case where the desired response is channel 1 (corresponding to stimulus at location S_1). Originally, we notice that stimulation at electrode $E_{1,2}$ causes the correct neuron to fire with high activation (top trace). Next, we notice degradation of the signal (after operation for 3 hours – middle trace). The degradation is such that on most occasions the output is incorrect. Finally, after searching and finding a better stimulation site at electrode $E_{2,3}$, we notice restoration of functionality to original levels (top trace). **(B)** Example of alteration of response to stimulation at an S_1 electrode in chamber A ($E_{1,2}$). Over time the distribution of spikes becomes distorted. Shifting the stimulus site to electrode $E_{2,3}$ partially restores the original statistics of the input signal.