









# Binding events through the mutual synchronization of spintronic nano-neurons

Miguel Romera <sup>1,2,3,7</sup>, Philippe Talatchian<sup>1,4,7</sup>, Sumito Tsunegi <sup>5</sup>, Kay Yakushiji <sup>5</sup>, Akio Fukushima<sup>5</sup>, Hitoshi Kubota <sup>5</sup>, Shinji Yuasa<sup>5</sup>, Vincent Cros <sup>1</sup>, Paolo Bortolotti<sup>1</sup>, Maxence Ernout<sup>1,6</sup>, Damien Querlioz<sup>6</sup>  & Julie Grollier <sup>1</sup> 

The brain naturally binds events from different sources in unique concepts. It is hypothesized that this process occurs through the transient mutual synchronization of neurons located in different regions of the brain when the stimulus is presented. This mechanism of ‘binding through synchronization’ can be directly implemented in neural networks composed of coupled oscillators. To do so, the oscillators must be able to mutually synchronize for the range of inputs corresponding to a single class, and otherwise remain desynchronized. Here we show that the outstanding ability of spintronic nano-oscillators to mutually synchronize and the possibility to precisely control the occurrence of mutual synchronization by tuning the oscillator frequencies over wide ranges allows pattern recognition. We demonstrate experimentally on a simple task that three spintronic nano-oscillators can bind consecutive events and thus recognize and distinguish temporal sequences. This work is a step forward in the construction of neural networks that exploit the non-linear dynamic properties of their components to perform brain-inspired computations.

<sup>1</sup>Unité Mixte de Physique, CNRS, Thales, Université Paris-Saclay, 91767 Palaiseau, France. <sup>2</sup>GFMC, Departamento de Física de Materiales, Universidad Complutense de Madrid, 28040 Madrid, Spain. <sup>3</sup>Unidad Asociada UCM/CSIC, Laboratorio de Heteroestructuras con Aplicación en Espintrónica, 28049 Madrid, Spain. <sup>4</sup>Université Grenoble Alpes, CEA, CNRS, Grenoble INP, SPINTEC, 38000 Grenoble, France. <sup>5</sup>National Institute of Advanced Industrial Science and Technology (AIST), Spintronics Research Center, Tsukuba, Ibaraki 305-8568, Japan. <sup>6</sup>Université Paris-Saclay, CNRS, Centre de Nanosciences et de Nanotechnologies, 91120 Palaiseau, France. <sup>7</sup>These authors contributed equally: Miguel Romera, Philippe Talatchian. ✉email: [damien.querlioz@universite-paris-saclay.fr](mailto:damien.querlioz@universite-paris-saclay.fr); [julie.grollier@cnrs-thales.fr](mailto:julie.grollier@cnrs-thales.fr)

Spintronic oscillators are nanoscale devices realized with magnetic tunnel junctions which have the potential to be integrated by hundreds of millions in electronic chips<sup>1</sup>. The microwave voltages that they produce have varying amplitude and frequency in response to direct current inputs. Their non-linear dynamical properties are rich and tunable, and can be leveraged to imitate different features of biological neurons, which makes them particularly promising for neuromorphic computing<sup>2–6</sup>. The transient dynamics of a single spintronic nano-oscillator has been used to implement reservoir computing, achieving state-of-the-art results on a simple spoken digit recognition task<sup>7,8</sup>. Four spintronic nano-oscillators have been trained to classify spoken vowels by phase-locking their oscillations to the strong input signals produced by external microwave sources<sup>9</sup>.

It is now essential to demonstrate that the mutual synchronization of spintronic nano-oscillators can be exploited for computing. Larger hardware networks of oscillators can be built if the oscillators directly influence each other and synchronize through the weak signals that they emit, without the need for high power-consumption amplification stages. The latter is possible with spin-torque nano-oscillators due to their outstanding synchronization ranges, enhanced by factors of typically ten compared to Kuramoto model-like phase oscillators due to the coupling between their amplitude and phase<sup>10–16</sup>. In addition, more complex tasks can be achieved by exploiting the rich interactions that emerge in assemblies of mutually coupled oscillators rather than using phase-locking to external signals<sup>17</sup>.

A primary source of inspiration to move in this direction is neuroscience, which shows that in the brain, vast groups of neurons mutually synchronize in response to external or internal stimuli, giving rise to strong oscillatory signals<sup>18–20</sup>. This process is often hypothesized to enable spatiotemporal integration of stimuli, a mechanism called ‘binding through synchrony’<sup>21</sup>. Neuroscience-inspired algorithms show that it can be used for pattern recognition in oscillatory neural networks if mutual synchronization can be controlled and tuned to achieve the desired task<sup>20–22</sup>.

In this work, we show that spintronic nano-oscillators can recognize temporal patterns through their mutual synchronization by binding together consecutive events in time. We implement a hardware neural network based on these principles with three spintronic nano-oscillators. We demonstrate that it recognizes sequences of spikes from a neuroscience-inspired database with a success rate of 94%, approaching the success rate of 96% achieved by identical and noiseless oscillators. We show that these high recognition rates stem from the possibility to precisely control the mutual synchronization of spintronic nano-oscillators by varying their frequency over large direct current input ranges.

## Results

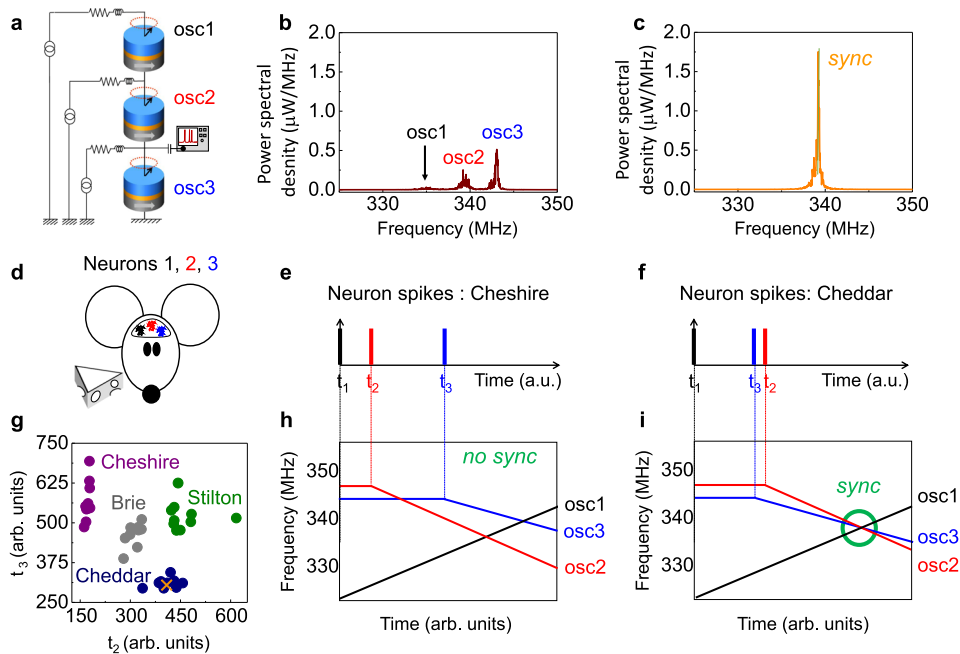
Our experiment exploits the coupling that occurs naturally when hardware spintronic nano-oscillators are electrically connected to synchronize them with each other<sup>10</sup>. The set-up is shown in Fig. 1a (details on samples and set-up are given in the ‘Methods’ section). An important feature of these nano-oscillators is that their frequency can be individually and easily controlled by varying the direct current through each oscillator. When their frequencies are well separated, the three oscillators emit microwave signals independently, and the spectrum analyzer at the output of the set-up displays three peaks (Fig. 1b and Supplementary Fig. 1). The propagation of these microwave emissions in the line creates a coupling between the connected oscillators. When the frequencies of the oscillators are brought closer together, within the mutual locking range—here of 5 MHz—they

synchronize, which results in the single peak of Fig. 1c. Its power is significantly higher than the sum of the individual emitted powers of the three disconnected oscillators measured with the same bias conditions (Supplementary Figs. 2, 3). This distinctive feature is due to the phase coherence between oscillators characteristic of the synchronized state (see ‘Methods’)<sup>14,15</sup>.

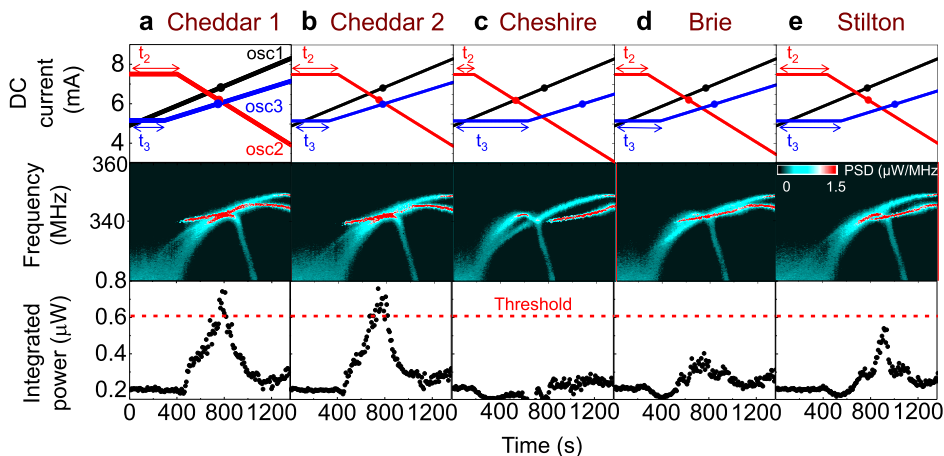
We leverage this mutual synchronization phenomenon to recognize temporal sequences with a hardware neural network of three spintronic oscillators. We consider a fictitious task (the construction of the dataset is described in ‘Methods’), in which a mouse is presented with four different categories of cheese (Fig. 1d). Each kind of cheese generates different neuron activities in the mouse brain. The temporal sequences to be classified are composed of three spikes, each recorded from three different neurons of the fictitious mouse brain. Figure 1e, f illustrates sequences of spikes when the mouse is in the presence of Cheshire and Cheddar cheese. By convention, the time of the first spike is taken as zero and the times of spikes for neurons 2 and 3 are shown in Fig. 1g for the whole database. The different colors correspond to different categories of cheese. Ten different samples of cheese per category are presented to the mouse, giving rise to the variability within each category. The goal of the task is to infer which type of cheese is presented to the mouse by analyzing the recorded sequence of spikes.

For this purpose, we use a mechanism initially proposed by Hopfield et al. to bind temporal features through neural synchronization<sup>21</sup>. A network is composed of as many neurons as there are spikes in the input sequence, and is trained to recognize a single category of input. Our dataset features three spikes, therefore we use three spintronic oscillators tuned to recognize a given kind of cheese, e.g. Cheddar. Each spike triggers a current ramp in the associated neuron. Figure 1h is a schematic illustrating the behavior of the network activated by a spike sequence that it has not been designed to recognize, e.g. Cheshire cheese. In that case, the ramps do not intersect at any time. On the contrary, when the oscillator network is activated with the spike sequence for Cheddar cheese that it has been designed to recognize, the different ramps intersect at a specific time, as illustrated in Fig. 1i. The neurons therefore transiently mutually synchronize and give rise to a large output signal, as in Fig. 1c signaling that they have bonded the events together and identified the sequence as meaningful.

In this framework, training the network means finding parameters for the ramps leading to mutual synchronization of the three oscillators for all the points of the database corresponding to the Cheddar category, even if the corresponding spike times are scattered in time. The detailed training procedure is described in ‘Methods’, and we present it succinctly here. We start by picking random values for the slopes of the ramps. For choosing the initial values of the currents flowing through each oscillator, we use inputs corresponding to the center of the ‘Cheddar’ data points cloud, pinpointed with an orange cross in Fig. 1g. We convert the arbitrary units of Fig. 1g to seconds: the ramps of current in oscillator 2 and oscillator 3 are therefore triggered with delays of 412 s and 308 s, respectively, for this calibration point. We then choose the oscillator initial frequencies so that the application of this sequence of spikes eventually leads to a set of currents  $I_{\text{Synch}} = (6.8 \text{ mA}, 6.2 \text{ mA}, 6.0 \text{ mA})$  for which the oscillators synchronize (see ‘Methods’). We then present the actual data points of Fig. 1g to the oscillators network. When the experiment behaves as expected (i.e., reaches synchronization when a ‘Cheddar’ data point is presented, or does not reach synchronization when another cheese point is presented), nothing is done. On the other hand, if the experiment does not behave as expected, the ramp slopes are corrected following the simple automatic procedure described in ‘Methods’, and the initial



**Fig. 1** Binding temporal sequences through synchrony. **a** Schematic of the experimental set-up with three spin-torque nano-oscillators electrically connected and coupled through the microwave currents they emit. **b, c** Microwave output emitted by the network of three coupled oscillators when they are not (**b**) and when they are (**c**) synchronized. **d** Schematic of the fictitious mouse to which four different categories of cheese are presented. Each category generates different activities in the three neurons of the mouse brain. **e, f** Sequence example of neuron spikes in the mouse brain in the presence of a piece of Cheshire (**e**) and Cheddar (**f**), respectively. **g** Inputs applied to the network, represented as the time of spikes for neurons 2 and 3. The spike of neuron 1 is set as the origin of the sequence and taken as zero. Each color corresponds to a different cheese category, and each data point corresponds to a different piece of cheese. **h, i** Ramps of current generated in the network upon application of the input spike sequences described in (**e**) and (**f**), corresponding to the presentation to the mouse of a piece of Cheshire and Cheddar, respectively, when the network is trained to recognize Cheddar.



**Fig. 2** Three hardware spintronic nano-oscillators recognize Cheddar. Response of the oscillator network trained to recognize Cheddar to spikes sequences generated in the mouse brain when a piece of Cheddar (**a, b**) or a different cheese (**c-e**) is presented. The spike sequences generate ramps of currents (top) which translate into variations of the oscillators frequencies (middle) and the network total emitted power (bottom). If a piece of Cheddar is presented (**a, b**), the ramps of current and their associated variations of frequencies lead to transient mutual synchronization of the three oscillators. This translates into an enhancement of the network total emitted power above the threshold shown in red dotted line (**a, b**, bottom) meaning recognition. If a different cheese is presented (**c-e**), the ramps of current do not give rise to the mutual synchronization of the three oscillators, and the total emitted power remains well below the threshold (**c-e**, bottom), meaning that the network distinguishes that the cheese presented is not Cheddar.

currents are recomputed so that the center of the ‘Cheddar’ data points cloud still leads to synchronization.

At the end of the learning procedure, the initial current conditions reach  $(I_{osc1}^0, I_{osc2}^0, I_{osc3}^0) = (4.9 \text{ mA}, 7.51 \text{ mA}, 5.15 \text{ mA})$  and the slopes of the ramps of current  $(dI_{osc1}/dt, dI_{osc2}/dt, dI_{osc3}/dt) = (2.5 \text{ } \mu\text{A/s}, -3.75 \text{ } \mu\text{A/s}, 1.875 \text{ } \mu\text{A/s})$ . These values are then maintained, and the experiment can be used to recognize new data points.

Figure 2a shows the measured oscillator responses to the sequence ‘Cheddar 1’ = (400 s, 294 s), which is the first entry in the class ‘Cheddar’. The corresponding trained ramps of currents are triggered at  $t_1 = 0, t_2 = 400 \text{ s},$  and  $t_3 = 294 \text{ s}$  in oscillator 1, 2, and 3, respectively. The three large dots in the top panel of Fig. 2 highlight the set of currents  $I_{synch}$  that we have used to calibrate the network, at which the oscillators synchronize. As can be seen

in Fig. 2a, after  $\sim 760$  s, the direct currents flowing through the oscillators reach values approaching  $I_{\text{synch}}$ . At this point, the frequencies of all three oscillators become near-identical (Fig. 2a-middle), and the total emitted power peaks at  $P_{\text{max}} = 0.743 \mu\text{W}$ . To know if the oscillators have recognized the category Cheddar, we need to assess if they have transiently synchronized or not. For this purpose, we can compare  $P_{\text{max}}(\mathbf{I})$  to the sum of the powers of the three independent oscillators for the same current conditions  $P_{\text{unsync}}(\mathbf{I})$ , equal to  $0.43 \mu\text{W}$  at this particular point. Here  $P_{\text{max}}$  is much larger than  $P_{\text{unsync}}$ , which shows that the oscillators have mutually synchronized and successfully recognized that the piece of cheese belongs to the Cheddar category.

The value of  $P_{\text{unsync}}$  varies strongly with the currents  $\mathbf{I}$  applied to the oscillators which makes it difficult to use it as a general criterion to detect synchronization for all sets of currents in the experiment. In the following, we consider that the network recognizes the input pattern if the total emitted power reaches a fixed threshold value of  $0.608 \mu\text{W}$ , independent of the current values (dotted red line in the bottom panel of Fig. 2) and if the frequencies of the three oscillators at that point are all equal (see “Methods”). Figure 2b shows the network response to another input from the Cheddar class. For this sequence ‘Cheddar2’ = (392 s, 315 s), the triggered ramps of current never reach  $I_{\text{synch}}$ , but they get close to this value around  $t = 744$  s, where a mutual synchronization event is observed, with the emission of a single peak accompanied with a large increase of the emitted power above the threshold. This input is therefore correctly classified as belonging to the category ‘Cheddar’. This example illustrates well the interest of using mutual synchronization to categorize spread data: synchronization occurs when oscillators frequencies are close, but they do not need to be exactly identical (as in Supplementary Fig. 1).

In contrast, Fig. 2c–e shows the network response to inputs of the other cheese categories, which characteristic time sequences never lead to sets of currents close to  $I_{\text{synch}}$ . In consequence, the three oscillators do not synchronize, and the power emitted by the network remains well below the recognition threshold. In these three cases (Fig. 2c–e) the network interprets correctly that the applied input does not correspond to the class ‘Cheddar’.

Table 1 shows the overall classification performances of the network. As can be seen in the first row, the network classifies correctly 9 out of 10 inputs corresponding to the class ‘Cheddar’. Moreover, when inputs from other categories (‘Cheshire’, ‘Brie’, and ‘Stilton’) are applied to the system, the network correctly interprets that the input is not a piece of ‘Cheddar’. The overall success rate for this category is 97.5%.

The same three oscillators can also be trained to recognize the other categories of cheese. For this, we modify the initial conditions of the network (the initial currents flowing through each oscillator  $I_{\text{osci}}^0$ , and the slopes of the ramps of current triggered

when an input is applied,  $dI_{\text{osci}}/dt$ ) so that the oscillator currents reach  $I_{\text{synch}}$  only when the desired category of input is applied to the system, following the same procedure as in the ‘Cheddar’ example (see “Methods”). Using these new calibration parameters, we repeat the recognition experiment by applying the same dataset as previously. The results are shown in Table 1: the network recognition rate for ‘Stilton’, ‘Brie’, and ‘Cheshire’ is, respectively, 95%, 92.5%, and 92.5%. Overall, the network responds correctly to 94% of the inputs. By comparison, a perceptron trained on the same database achieves a recognition rate of 93.3% (see “Methods”). Supplementary Fig. 4 compares the performance of a simulated network of three identical, noiseless oscillators to the perceptron for increased spike jitter in the database. Interestingly, the ideal oscillator network does better than the perceptron on moderate jitter values. For higher jitter values (standard deviation of the jitter of 50 time units and higher), the perceptron outperforms the oscillator network. This result, promising for applications, therefore also leaves room for improvement by optimizing the learning algorithm.

## Discussion

The excellent performance of this simple network comes from the match between the requirements of the algorithm and the physical properties of spintronic nano-oscillators. Here the inputs are sequences of events that are largely spread in time and need to be bonded to constitute a single concept. The algorithm does this task in two steps. First, it converts the spread timing of events in the sequence to close-by neuron frequencies by ramping the values of frequencies over wide ranges. The high-frequency tunability of spintronic nano-oscillators<sup>10</sup> provides a straightforward hardware implementation of this property. Second, the algorithm leverages neuron synchronization to bind these neighboring frequencies into a single concept, here, a cheese category. This synchronization range must be large enough to ensure that events are bonded even if different sequences encoding the same category of cheese are scattered in time. The large synchronization bandwidths accessible to spintronic nano-oscillators<sup>10</sup> reduce the precision requirements on the frequency, and therefore on the direct current steps, needed to achieve ramp convergence.

Complementary metal-oxide semiconductor (CMOS) technology-based voltage-controlled oscillators such as ring oscillators can also exhibit such characteristics, but large-area capacitors are needed to control their synchronization<sup>23</sup>. The total silicon area occupied by a single oscillator is larger than  $3 \times 3 \mu\text{m}^2$ <sup>24</sup>, whereas spin-torque nano-oscillators can be scaled below  $100 \times 100 \text{nm}^2$ . Using small-area oscillators is imperative, as moving beyond toy tasks will require scaling up the system. Our simple network processes inputs composed of three events, but more complex inputs will be based on a larger number of events and will require more oscillators (see “Methods”).

The original paper of Hopfield uses 40 neurons for recognizing the first ten spoken digits<sup>21</sup>. It has been shown recently that such large numbers of spintronic nano-oscillators can mutually synchronize, by driving them with spin-Hall torques and coupling them strongly through ferromagnetic exchange<sup>12</sup>. Their individual frequencies can be controlled by tuning the current or magnetic properties in the nano-constriction region<sup>25</sup>. To confirm this possibility and evaluate the scalability of our approach to larger scale networks, we simulated a network of  $N = 90$  coupled spin-torque nano-oscillators, and we compare its performance classifying real biological data with the one of a perceptron of the same size. These results are described in Supplementary Note 1, Supplementary Fig. 5, and Supplementary Table 5. Interestingly, the oscillators network performs better than the perceptron on this task.

**Table 1 Recognition rates.**

Cheese to be detected	Presented cheese (10 data points)				Recognition rate
	Cheddar	Stilton	Brie	Cheshire	
	Number of recognitions (out of 10)				
Cheddar	9	0	0	0	97.5%
Stilton	0	8	0	0	95%
Brie	0	3	10	0	92.5%
Cheshire	0	0	0	7	92.5%

Number of recognitions out of 10 presented samples of each cheese, when the network is trained to classify Cheddar, Brie, Cheshire, and Stilton, respectively. The column “Recognition rate” refers to the percentage of times that the network responds correctly, either because it detects that the input belongs to the category it was trained to recognize or because it interprets correctly that the inputs correspond to another cheese category.

Important for scaling to even larger dimensions, the algorithm is very tolerant to device variability: as long as the oscillators synchronize over large bandwidths, current ramps can be found for every input through training<sup>21</sup>. Recognition is achieved at the timescale of tens of seconds or minutes in our proof of concept, but can potentially be sped up to a few tens of nanoseconds, as spintronic nano-oscillators can be tuned and synchronized within these timescales<sup>26,27</sup>. Finally, when the oscillators synchronize, they generate an additional direct voltage through an effect called spin-diode, that can be detected and then processed with simple and energy-efficient CMOS circuits, as described in the Methods section of ref. <sup>9</sup>. This work therefore constitutes a milestone towards the implementation of large-scale oscillatory neural networks using the physical properties of spintronic nano-oscillators to compute.

## Methods

**Samples.** Magnetic tunnel junctions (MTJs) with a structure of buffer/PtMn(15)/Co<sub>71</sub>Fe<sub>29</sub>(2.5)/Ru(0.9)/Co<sub>60</sub>Fe<sub>20</sub>B<sub>20</sub>(1.6)/Co<sub>70</sub>Fe<sub>30</sub>(0.8)/MgO(1)/Fe<sub>80</sub>B<sub>20</sub>(6)/MgO(1)/Ta(8)/Ru(7) (thicknesses in nm) were deposited by ultrahigh-vacuum (UHV) magnetron sputtering. After annealing at 360 °C for 1 h, the resistance-area product was RA ≈ 3.6 Ω μm<sup>2</sup>. Circular-shaped MTJs with a diameter of about 375 nm were patterned using Ar ion etching and e-beam lithography. The resistance of the devices is close to 40 Ω, and the magneto-resistance ratio is about 100% at room temperature. For the dimensions used here, the FeB layer presents a magnetic vortex as the ground state. In the vortex core (a small region of about 12 nm diameter at remanence for our materials), the magnetization spirals out of plane. Under direct current injection and the action of the spin-transfer torques, the core of the vortex steadily gyrates around the center of the dot with a frequency in the range of 150 to 450 MHz for the oscillators we used here.

**Experimental set-up.** Figure 1a shows a schematic of the experimental set-up with three electrically coupled vortex nano-oscillators. A magnetic field of μ<sub>0</sub>H = 400 mT is applied perpendicularly to the oscillator layers to tilt the magnetization of the polarizer out-of-plane so that the spin-transfer torque acting on the vortex can compensate for the damping torque and induce self-sustained oscillations<sup>28</sup>. A direct current is injected into each oscillator to induce vortex dynamics, which leads to periodic oscillations of the magneto-resistance, giving rise to an oscillating voltage at the same frequency than the vortex core dynamics. The three oscillators are electrically connected by millimeter-long wires. In this configuration, the microwave current generated by each oscillator propagates in the electrical microwave circuit, influencing the dynamic and, in particular, the frequency of the other oscillators through the microwave spin-torques it creates. The oscillators are therefore electrically coupled through the microwave currents they emit, and too far away to be coupled through the magnetic dipolar fields that they radiate. Three direct currents (I<sub>DC1</sub>, I<sub>DC2</sub>, I<sub>DC3</sub>) are supplied to the circuit by three different sources. The actual current flowing through each spin-torque oscillator is given by I<sub>STO1</sub> = I<sub>DC1</sub>, I<sub>STO2</sub> = I<sub>DC2</sub> + I<sub>DC1</sub>, and I<sub>STO3</sub> = I<sub>DC3</sub> + I<sub>DC2</sub> + I<sub>DC1</sub>, respectively, where I<sub>STO*i*</sub> corresponds to the current flowing through the *i*th oscillator. Therefore, the current flowing through each oscillator can be tuned independently by controlling the three applied direct currents (I<sub>DC1</sub>, I<sub>DC2</sub>, I<sub>DC3</sub>) at the same time. Thus, we can control the frequency of each oscillator independently. This approach is scalable to several hundreds of oscillators, as shown in ref. <sup>29</sup>. The microwave signal emitted by the coupled system is recorded by a spectrum analyzer. Color maps showing the evolution of frequency and microwave power emitted by each individual oscillator as a function of current and field are displayed in Supplementary Fig. 1.

**Synchronization detection.** When spin-torque oscillators mutually synchronize their non-linear magnetization dynamics reaches a new state characterized by the oscillators phases being locked to each other. This stabilizes their oscillations frequencies and reduces the main sources of noise in the magnetization dynamics: the amplitude noise and the phase noise, the latter being particularly disruptive for the oscillations coherence. In consequence, the signature of mutual synchronization state is an emission spectrum which shows a drastic increase of the spectral coherence. This is characterized by an emitted power which is above the sum of the powers emitted by the individual oscillators when they are not coupled<sup>12,25,30</sup>.

Supplementary Fig. 2 shows the emitted spectra of the three oscillators under the same conditions of applied current as in Fig. 1c but when they are not connected to each other. As can be seen, the frequencies are close to each other, but they are not equal. This result illustrates well the interest of using mutual synchronization to categorize spread data: oscillators frequencies do not need to be exactly identical to observe mutual synchronization, as soon as they are closer than the locking range.

Supplementary Fig. 3 shows the oscillators network emitted power (black dots) during the experiment shown in Fig. 2a. The total emitted power reaches a

maximum value of P<sub>max</sub>(I) = 0.743 μW. This value is significantly higher than the sum of the individual emitted powers of the three oscillators (dash red line), which is around P<sub>unsync</sub>(I) = 0.43 μW at its maxima in this particular experiment.

The value of P<sub>unsync</sub> varies strongly with the currents I applied to the oscillators. If the current slopes in all oscillators are chosen positive, the value of P<sub>unsync</sub> increases with time. A synchronization event happening at the beginning of the ramp (at low time and low applied currents) may emit similar power than the array of unsynchronized oscillators at the end of the ramps (at large currents). This would require defining a time-dependent power threshold to detect synchronization.

By choosing different signs for the current slopes in the oscillators this situation can be avoided and we can use a constant power threshold for synchronization detection. We consider recognition when the total emitted power overcomes a threshold value of 0.608 μW and the oscillators have the same frequency at that point. The threshold value has been chosen to minimize misclassification errors, considering the emitted power of the independent oscillators at all possible applied currents within the experimental range.

**Construction of the cheese dataset.** The artificial dataset shown in Fig. 1g was constructed based on the dataset of formants of spoken vowels, available publicly as Supplementary Data in ref. <sup>9</sup>, rescaled by hand to reach delay values in the scale of our experimental set-up. The delay t<sub>2</sub> is equal to 4.8 × 10<sup>5</sup>/f<sub>2</sub>, and t<sub>3</sub> = 2.4 × 10<sup>5</sup>/f<sub>1</sub>, where f<sub>1</sub> and f<sub>2</sub> are the two first formants of the vowels. The delay t<sub>1</sub> is always equal to zero. The different “cheeses” correspond to different vowels, and in each case, we use the first ten samples of the dataset. “Cheshire” corresponds to /iy/, “Brie” to /er/, “Stilton” to /uw/, and “Cheddar” to /aw/.

## Trained initial conditions to classify each category of cheese/calibration parameters.

The network of oscillators can learn to classify new data. Here, training means finding the parameters of the ramps of current (i.e., initial currents flowing through each oscillator and slopes of the ramps triggered when an input is applied) that lead to mutual synchronization (recognition) when a particular class of input is applied. To do so, we first identified conditions of current at which the three oscillators mutually synchronize: I<sub>Synch</sub> = (6.8 mA, 6.2 mA, 6.0 mA).

The first values of the three slopes of the current ramps are then chosen randomly, between values 2.5 and 4 μA/s. The initial values of the currents flowing through each oscillator are then calculated, using delays corresponding to the center of the cloud of each category in Fig. 1g: the initial currents are chosen such as if a data point corresponding exactly to the center of the cloud of the trained category is applied, the oscillator currents reach exactly I<sub>Synch</sub>.

Then, the following empirical training process is used to optimize the values of the slopes of the current ramps. Data points corresponding to different cheese are presented to the system. For each data point, if the system provides a correct response (mutual synchronization if the right cheese was presented, no synchronization if the wrong cheese was presented), no change to the parameters is done. Conversely, if the system was expected to reach mutual synchronization and did not, the value of the maximum slope (in absolute value) is reduced by an hyperparameter Δ. If the system was not expected to reach mutual synchronization and still synchronized, the minimum slope (in absolute value) is increased by the hyperparameter Δ. After each update of the slopes of the current ramps, the initial currents are recomputed using the same method as initially: they are chosen so that if a data point corresponding exactly to the center of the cloud of a category is applied, the oscillator currents reach exactly I<sub>Synch</sub>.

In our experiments, to simplify the training, we kept the ramp of oscillator 1 fixed for all categories and varied only the ramps of oscillators 2 and 3. Supplementary Table 1 shows the trained calibration parameters used for each input that the network is trained to classify.

**Simulations with ideal oscillators.** The pattern recognition scheme of the experiment was simulated with a network of three identical noiseless oscillators, using the database of Fig. 1g. The only parameter that differs from one simulated oscillator *i* from the other one is its applied direct current I<sub>*i*</sub>. The simulated oscillators correspond to vortex-based spin-torque oscillators as in the experiment. Their dynamics follows the differential Thiele equation model:

$$\mathbf{G}_i \times \frac{d\mathbf{X}_i}{dt} - \hat{D}_i(\mathbf{X}_i) \frac{d\mathbf{X}_i}{dt} - \frac{\partial W_i(\mathbf{X}_i, I_i, I_{com}^f)}{\partial \mathbf{X}_i} + \mathbf{F}_i^{STT}(\mathbf{X}_i, I_i, I_{com}^f) = \mathbf{0} \quad (1)$$

here,  $\mathbf{X}_i = \begin{pmatrix} x_i \\ y_i \end{pmatrix}$  is the vortex core position,  $\mathbf{G}_i$  is the gyrovector,  $\hat{D}_i$  is the damping,

$W_i$  is the potential energy of the vortex,  $\mathbf{F}_i^{STT}$  is the spin-transfer force, and  $I_{com}^f$  is a common microwave current. This model reproduces well micromagnetic simulations<sup>13</sup>. It also successfully describes experimental results with spin-torque nano-oscillators and can easily be generalized to non-linear auto-oscillators as van der Pol oscillators<sup>9</sup>. The parameters used for the Thiele equation in the simulation are expressed in Supplementary Table 2.

A fourth-order Runge-Kutta scheme is used to solve simultaneously the three coupled differential Thiele equations corresponding to the three coupled oscillators. The integration time step was set to 0.01 ns. Simulations are achieved at T = 0 K (no thermal noise). As in the experiment, the simulated oscillators are

electrically coupled through the sum of their individual microwave alternative current emissions. The expression of this common current  $I_{\text{com}}^f$  is described as follows<sup>9</sup>:

$$I_{\text{com}}^f = \frac{1}{Z_0 + \sum_{i=1}^3 R_i} \left( \sum_{i=1}^3 \lambda \Delta R_i I_i y_i \right) \quad (2)$$

here  $\Delta R_i$  is the mean resistance variation caused by the vortex core gyrotropic motion,  $I_i$  is the direct current flowing through the  $i$ -th oscillator,  $R_i$  is its mean electrical resistance,  $Z_0 = 50 \Omega$  is the load impedance and  $\lambda = \frac{2}{3}$ .

The main variables extracted from the simulations are the three steady-state frequency ( $f_1$ ,  $f_2$ ,  $f_3$ ) of the three oscillators obtained at a given set of direct currents ( $I_1$ ,  $I_2$ ,  $I_3$ ). As in the experiment, a given direct current set ( $I_1$ ,  $I_2$ ,  $I_3$ ) corresponds to a step of the current ramp of the network. In order to extract ( $f_1$ ,  $f_2$ ,  $f_3$ ) from the simulations, first the instantaneous frequency of each oscillator is determined through the simulated cartesian trajectory and velocity of the vortex core over  $5 \mu\text{s}$ . Then, the steady-state frequency is computed by evaluating the temporal average of the instantaneous frequency over only the last 60% of the simulated time trace corresponding to  $3 \mu\text{s}$ . Due to the electrical coupling, these steady-state frequencies differ from those obtained in the individual uncoupled case. Depending on the direct current received by each oscillator, their frequencies are pulled and can eventually merge leading to a mutual synchronization. As in experiments, a recognition event corresponds to a mutual synchronization of all the three oscillators to a common frequency. In order to systematically detect this type of events in simulations, we analyze the frequency difference between the three oscillators as follows:

- If  $|f_1 - f_2| \leq f_{\text{th}}$  and  $|f_2 - f_3| \leq f_{\text{th}}$  then the three oscillators are mutually synchronized
- Otherwise, the three oscillators are considered to be not synchronized all together.

Here  $f_{\text{th}}$  is a threshold value set to 0.1 MHz. Following criteria (a) and (b), if the three oscillators remain mutually synchronized for at least three consecutive direct current ramp steps corresponding to at least  $13 \mu\text{s}$ , then the simulated network is considered to be in a recognition state. The initial value  $I^0$  and slope value  $dI/dt$  of the direct current ramps in simulations are calibrated following the method described in the section “Trained initial conditions to classify each category of cheese/Calibration parameters”. Supplementary Table 3 shows the trained calibration parameters used for each input that the network is trained to classify. Following the same procedure used in experiments, the simulated recognition performances are evaluated for each class of cheese using the associated initial conditions (Supplementary Table 3). The recognition rates obtained through this procedure are shown in Supplementary Table 4. Overall, the network responds correctly to 96% of the inputs.

We have then investigated through simulations how robust is the oscillator network to increased spike jitter and compared it to the performance of a perceptron trained on the same data.

We have generated three new databases that keep the same asymmetry as the initial database but a larger jitter in each category. To generate these three new datasets, we use the initial data points that do not have jitter ( $t_2$ ,  $t_3$ ) (see Fig. 1g). For each of these points, we construct 10 new points ( $t_2'$ ,  $t_3'$ ) for which the coordinates were randomly chosen with a normal distribution that is centered at the initial point ( $t_2$ ,  $t_3$ ) and has a fixed standard deviation. To increase the jitter, we increased the standard deviation of the normal distribution used to construct ( $t_2'$ ,  $t_3'$ ) points. More precisely, we choose three different values of standard deviation (10, 50, and 100), which give rise to the three new databases represented in Supplementary Fig. 4a. The data we used to train and evaluate the performance of the simulated oscillator network consists of two-dimensional vectors  $x_{\text{input}} = (t_2', t_3')$ . Similarly, the data we used to train and evaluate the performance of the perceptron model consists of two-dimensional vectors  $x_{\text{input}} = (t_2'/400, t_3'/400)$ , in which each coordinate of the databases has been normalized by 400. Panel b of Supplementary Fig. 4 shows simulations of the oscillator network performance compared to a perceptron. Both networks are trained on the initial database (standard deviation of 0 in panel b) and tested on the new databases.

The perceptron consists of two input neurons and one output neuron, with two weights and one bias, i.e., three learnable parameters, as in the oscillators network. As in the experiments, a batch size of one is used (i.e., samples are presented one by one). We have performed learning on the four binary tasks “this cheese”/“not this cheese” for each type of cheese. The perceptron does not have hidden layers, and the neuron activation function at the output layer is a sigmoid function. If the output is equal to or larger than 0.5, the perceptron outputs “this cheese”, while if it is smaller than 0.5 the perceptron outputs “not this cheese”. We performed backpropagation (by stochastic gradient descent) over the negative binary log-likelihood (or binary cross-entropy) for training the network. Initial weights and biases were initialized randomly from a uniform distribution bounded between  $-1$  and  $1$ . To ensure convergence we used a learning rate of 0.1 during 100 iterations. As expected, the recognition performance drops with increased spike time jitter.

## Data availability

The datasets analyzed during this study are available as Supplementary Data in ref. <sup>9</sup> (see Methods). Experimental data are available from the authors under reasonable request.

## Code availability

The codes and simulation files that support the plots and data analysis within this paper are available from the authors on reasonable request.

Received: 5 June 2020; Accepted: 10 January 2022;

Published online: 15 February 2022

## References

- Golonzka, O. et al. MRAM as embedded non-volatile memory solution for 22FFL FinFET technology. in *2018 IEEE International Electron Devices Meeting (IEDM)* 18.1.1–18.1.4 (2018).
- Macià, F., Kent, A. D. & Hoppensteadt, F. C. Spin-wave interference patterns created by spin-torque nano-oscillators for memory and computation. *Nanotechnology* **22**, 095301 (2011).
- Pufall, M. R. et al. Physical implementation of coherently coupled oscillator networks. *IEEE J. Explor. Solid-State Comput. Devices Circuits* **1**, 76–84 (2015).
- Grollier, J., Querlioz, D. & Stiles, M. D. Spintronic nanodevices for bioinspired computing. *Proc. IEEE* **104**, 2024–2039 (2016).
- Cai, J. et al. Sparse neuromorphic computing based on spin-torque diodes. *Appl. Phys. Lett.* **114**, 192402 (2019).
- Csaba, G. & Porod, W. Coupled oscillators for computing: a review and perspective. *Appl. Phys. Rev.* **BIE2019**, 011302 (2020).
- Torrejon, J. et al. Neuromorphic computing with nanoscale spintronic oscillators. *Nature* **547**, 428–431 (2017).
- Tsunegi, S. et al. Physical reservoir computing based on spin torque oscillator with forced synchronization. *Appl. Phys. Lett.* **114**, 164101 (2019).
- Romera, M. et al. Vowel recognition with four coupled spin-torque nano-oscillators. *Nature* **563**, 230 (2018).
- Slavin, A. & Tiberkevich, V. Nonlinear auto-oscillator theory of microwave generation by spin-polarized current. *IEEE Trans. Magn.* **45**, 1875–1918 (2009).
- Houshang, A. et al. Spin-wave-beam driven synchronization of nanocontact spin-torque oscillators. *Nat. Nanotechnol.* **11**, 280–286 (2016).
- Awad, A. A. et al. Long-range mutual synchronization of spin Hall nano-oscillators. *Nat. Phys.* **13**, 292–299 (2017).
- Locatelli, N. et al. Efficient synchronization of dipolarly coupled vortex-based spin transfer nano-oscillators. *Sci. Rep.* **5**, 17039 (2015).
- Kaka, S. et al. Mutual phase-locking of microwave spin torque nano-oscillators. *Nature* **437**, 389–392 (2005).
- Mancoff, F. B., Rizzo, N. D., Engel, B. N. & Tehrani, S. Phase-locking in double-point-contact spin-transfer devices. *Nature* **437**, 393–395 (2005).
- Quinsat, M. et al. Amplitude and phase noise of magnetic tunnel junction oscillators. *Appl. Phys. Lett.* **97**, 182507 (2010).
- Vassilieva, E., Pinto, G., Barros, J. Ade & Suppes, P. Learning Pattern Recognition Through Quasi-Synchronization of Phase Oscillators. *IEEE Trans. Neural Netw.* **22**, 84–95 (2011).
- Uhlhaas, P. et al. Neural synchrony in cortical networks: history, concept and current status. *Front. Integr. Neurosci.* **3**, 17 (2009).
- Buzsáki, G. & Draguhn, A. Neuronal oscillations in cortical networks. *Science* **304**, 1926–1929 (2004).
- Hopfield, J. J. & Brody, C. D. What is a moment? “Cortical” sensory integration over a brief interval. *Proc. Natl Acad. Sci. USA* **97**, 13919–13924 (2000).
- Hopfield, J. J. & Brody, C. D. What is a moment? Transient synchrony as a collective mechanism for spatiotemporal integration. *Proc. Natl Acad. Sci. USA* **98**, 1282–1287 (2001).
- Ignatov, M., Ziegler, M., Hansen, M. & Kohlstedt, H. Memristive stochastic plasticity enables mimicking of neural synchrony: Memristive circuit emulates an optical illusion. *Sci. Adv.* **3**, e1700849 (2017).
- Nikonov, D. E. et al. Convolution inference via synchronization of a coupled CMOS oscillator array. *IEEE J. Explor. Solid-State Comput. Devices Circuits* **6**, 170–176 (2020).
- Nikonov, D. E. et al. A coupled CMOS oscillator array for 8 ns and 55 pJ inference in convolutional neural networks. Preprint at <https://arxiv.org/abs/1910.11803> (2019).
- Zahedinejad, M. et al. Two-dimensional mutually synchronized spin Hall nano-oscillator arrays for neuromorphic computing. *Nat. Nanotechnol.* **15**, 47–52 (2020).
- Manfrini, M. et al. Agility of vortex-based nanocontact spin torque oscillators. *Appl. Phys. Lett.* **95**, 192507 (2009).

27. Ruiz-Calaforra, A. et al. Frequency shift keying by current modulation in a MTJ-based STNO with high data rate. *Appl. Phys. Lett.* **111**, 082401 (2017).
28. Dussaux, A. et al. Large microwave generation from current-driven magnetic vortex oscillators in magnetic tunnel junctions. *Nat. Commun.* **1**, 8 (2010).
29. Talatchian, P. et al. Designing large arrays of interacting spin-torque nano-oscillators for microwave information processing. *Phys. Rev. Appl.* **13**, 024073 (2020).
30. Lebrun, R. et al. Mutual synchronization of spin torque nano-oscillators through a long-range and tunable electrical coupling scheme. *Nat. Commun.* **8**, 15825 (2017).

## Acknowledgements

This work was supported by the European Research Council ERC under Grant bioS-PINspired 682955. M.R. acknowledges financial support from Spanish MIC, AEI and FEDER through Grant No. PGC2018-099422-A-I00 (MCI/AEI/FEDER, UE) and from Comunidad de Madrid (Atracción de Talento Grant No. 2018-T1/IND-11935).

## Author contributions

The study was designed by J.G. and D.Q., samples were optimized and fabricated by S.T. and K.Y., experiments were performed by M.R. and P.T., numerical simulations were realized by P.T. and M.E. All authors contributed to analyzing the results and writing the paper.

## Competing interests

The authors declare no competing interests.

## Additional information

**Supplementary information** The online version contains supplementary material available at <https://doi.org/10.1038/s41467-022-28159-1>.

**Correspondence** and requests for materials should be addressed to Damien Querlioz or Julie Grollier.

**Peer review information** *Nature Communications* thanks the anonymous reviewer(s) for their contribution to the peer review of this work.

**Reprints and permission information** is available at <http://www.nature.com/reprints>

**Publisher's note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this license, visit <http://creativecommons.org/licenses/by/4.0/>.

© The Author(s) 2022