nature nanotechnology

LETTERS

https://doi.org/10.1038/s41565-021-00943-y

Check for updates

Scalable massively parallel computing using continuous-time data representation in nanoscale crossbar array

Cong Wang[1,3], Shi-Jun Liang[1,3], Chen-Yu Wang[1], Zai-Zheng Yang[1], Yingmeng Ge[2], Chen Pan[1], Xi Shen[1], Wei Wei[1], Yichen Zhao[1], Zaichen Zhang[2], Bin Cheng[1], Chuan Zhang[2] and Feng Miao[1]✉

The growth of connected intelligent devices in the Internet of Things has created a pressing need for real-time processing and understanding of large volumes of analogue data. The difficulty in boosting the computing speed renders digital computing unable to meet the demand for processing analogue information that is intrinsically continuous in magnitude and time. By utilizing a continuous data representation in a nanoscale crossbar array, parallel computing can be implemented for the direct processing of analogue information in real time. Here, we propose a scalable massively parallel computing scheme by exploiting a continuous-time data representation and frequency multiplexing in a nanoscale crossbar array. This computing scheme enables the parallel reading of stored data and the one-shot operation of matrix–matrix multiplications in the crossbar array. Furthermore, we achieve the one-shot recognition of 16 letter images based on two physically interconnected crossbar arrays and demonstrate that the processing and modulation of analogue information can be simultaneously performed in a memristive crossbar array.

Discrete-time and continuous-time signals are available for information representation in the time domain. A typical example is the binary data representation (that is, '1' and '0') used for computing in digital computers based on the von Neumann architecture, in which the bit stream is discrete in time and amplitude. The upper bound of the processor speed in digital computers is largely set by the clock frequency, the increase of which is essentially limited by the speed of flipping logic states. Further increasing the processor speed will lead to serious overheating issues[1,2], which explains why the clock frequency of advanced digital computers has stopped growing for over ten years[3]. This effect renders digital computing highly challenging in many applications, such as intelligent edge applications in the Internet of Things (IoT) network with the explosive growth of connected edge devices, which require high-efficiency data processing and communication[4–10].

Alternative computing schemes, other than digital computing, are thus required for these applications[5,11–23]. In contrast with discrete data representation, the use of a continuous-time data representation can avoid flips between different logic states and can overcome the aforementioned speed bottleneck of the processor[24]. Processing information represented with a continuous-time signal demands a nanoscale hardware architecture on which computation can be implemented in a continuous-time manner. Nanoscale memristive crossbars offer an ideal platform that can implement analogue computing[25–34], in which energy-efficient visual/speech processing and recognition have been achieved[35–39].

In this article, we propose and implement a scalable massively parallel computing scheme in a nanoscale crossbar array by employing a continuous-time data representation and frequency multiplexing and demonstrating its promising application in intelligent edge devices. As a demonstration, the proposed massively parallel computing allows the one-shot recognition of 16 letters in a neural network composed of two physically interconnected crossbar arrays. Moreover, massively parallel computing and signal modulation are implemented simultaneously in the analogue domain, opening up unprecedented opportunities for intelligent edge applications.

Figure 1 shows the continuous-time data representation and the corresponding continuous-time analogue computing scheme. This computing approach manifests itself in the data representation of continuous time and amplitude and takes full advantage of the physical attributes of the memristive crossbars to process data in a continuous-time domain, without suffering from issues associated with the steep rising and falling edges in digital computing. By employing Kirchhoff's current law and Ohm's law, any individual sinusoidal (or cosinusoidal) signals with different frequencies can be processed by the memristive crossbar array. Moreover, we can further feed a continuous-time signal by the linear combination of sinusoidal/cosinusoidal signals into the memristive crossbar array to effectively increase the computing capacity (Fig. 1a). As shown in the middle panels of Fig. 1a, such a continuous-time signal (for example, voltage or current) in a time segment can be transformed to a frequency spectrum with multiple peaks at different frequencies. Such a transformation explicitly illustrates that the input continuous-time voltage signals of single or multiple frequency components can be processed by the memristive crossbar array to output the continuous-time current signals of multiple frequency components. This transformation yields a unique process to read the information stored in the crossbar array and perform computing.

It is mathematically provable to implement matrix–matrix multiplication (MMM) by using the continuous-time computing scheme in a memristive crossbar array. The input voltage signal in the ith row (U_i) can be expanded by a series of orthogonal bases in the frequency domain as $U_i = \sum_k U_i^k$, where U_i^k is the voltage amplitude of the kth frequency. Similarly, the current output in the jth column

[1]Institute of Brain-Inspired Intelligence, National Laboratory of Solid State Microstructures, School of Physics, Collaborative Innovation Center of Advanced Microstructures, Nanjing University, Nanjing, China. [2]National Mobile Communications Research Laboratory, Frontiers Science Center for Mobile Information Communication and Security, Southeast University, Purple Mountain Laboratories, Nanjing, China. [3]These authors contributed equally: Cong Wang, Shi-Jun Liang. ✉e-mail: miao@nju.edu.cn

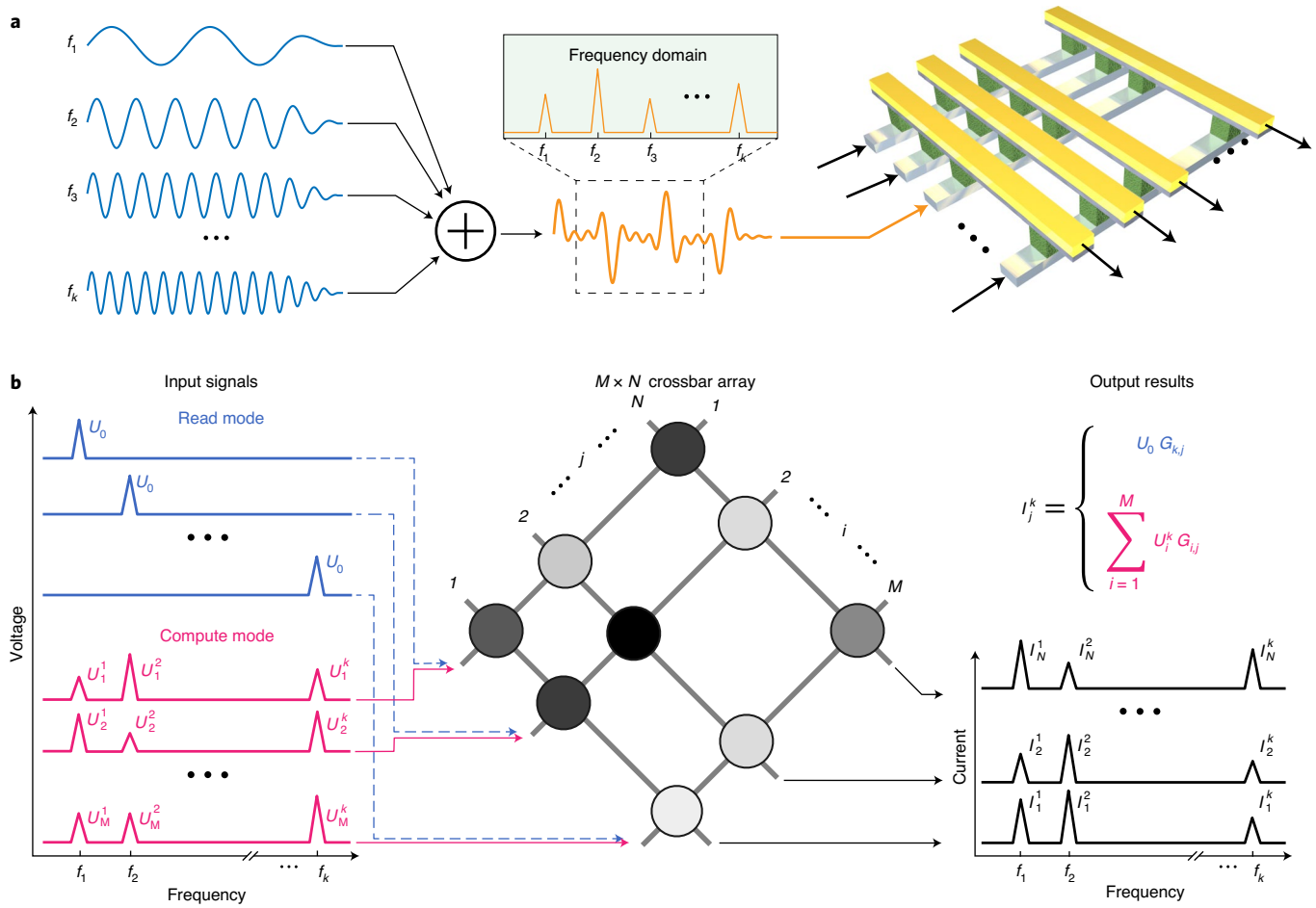


Fig. 1 | Continuous-time data representation for FMC in a memristive crossbar array. a, Implementation of FMC by using the memristive crossbar array, in which data are represented by a continuous-time signal synthesized with various sinusoidal (or cosinusoidal) signals with different frequencies f_k. **b**, Schematic illustration of FMC-enabled parallel reading and parallel computing. In Read mode, the parallel readout of all conductance values ($G_{k,j}$) stored in the memristive crossbar array can be achieved by feeding single-frequency continuous-time signals with a constant voltage amplitude U_0 into a row i (i=1, 2...M) of the crossbar array. In Compute mode, a one-shot MMM operation can be implemented by inputting multiple-frequency continuous-time signals with the voltage amplitude U_M^k at different frequency f_k values into row i (i=1, 2...M) of the crossbar array. For both Read and Compute modes, the output signals are generated from column j (j=1, 2…N) in the form of the current–frequency spectrum, in which I_j^k represents the current amplitude at the frequency component f_k of the jth column.

(I_j) can also be expanded as $I_j = \sum_k I_j^k$, where I_j^k is the current amplitude of the kth frequency. By assuming a constant conductance $G_{i,j}$ during operation, the input continuous-time voltage signals are converted to the output continuous-time current signal through Ohm's law $I_{i,j} = \sum_k U_i^k G_{i,j}$. The currents at all columns are summed according to Kirchhoff's current law $I_j = \sum_i I_{i,j}$. Since $I_j = \sum_k I_j^k$, we are able to achieve MMM through $I_j^k = \sum_i U_i^k G_{i,j}$ via a one-shot operation. Based on the continuous-time data representation, this frequency multiplexing computing (FMC) technology allows for the realization of massively parallel computing.

Implementing the one-shot MMM operation in the memristive crossbars enables massively parallel reading and computing, as schematically shown in Fig. 1b. The crossbar array can be operated in either Compute mode or Read mode, which is dependent on the frequency spectrum of the input continuous-time voltage signal. When the input signal fed into each row i (i=1, 2...M) of an $M \times N$ crossbar array contains a single-frequency component and a constant voltage amplitude U_0, the parallel reading of the data stored in the crossbar array is achievable. Meanwhile, when the input signal contains multiple frequency components with different voltage amplitudes, the crossbar array is capable of implementing massively parallel computing. Regardless of the Read mode or the Compute mode, the output results at each column of the crossbar array j (j=1, 2...N) are represented via the current–frequency spectrum.

We next implement these two FMC-based operation modes experimentally in a nanoscale memristive crossbar array. Figure 2a,b shows scanning electron microscopy images of a fabricated Ta/HfO_2 memristor crossbar array and the corresponding current–voltage (I–V) characteristics, respectively. All available resistive states in each device exhibit linear and symmetrical I–V characteristics within the voltage range from −50 mV to 50 mV. We also characterized the stability of conductance via 10^5 reading operations on memristive devices with resistance states from 1,000 to 10,000 ohms and present the corresponding error statistic in the top inset of Fig. 2b. The negligible standard deviation indicates the excellent stability of the fabricated memristive devices. To achieve the parallel reading of the stored data in the crossbar array, we applied continuous-time signals with the same voltage amplitude U_0 but distinct frequencies into each column of the crossbar array. Subsequently, we analysed

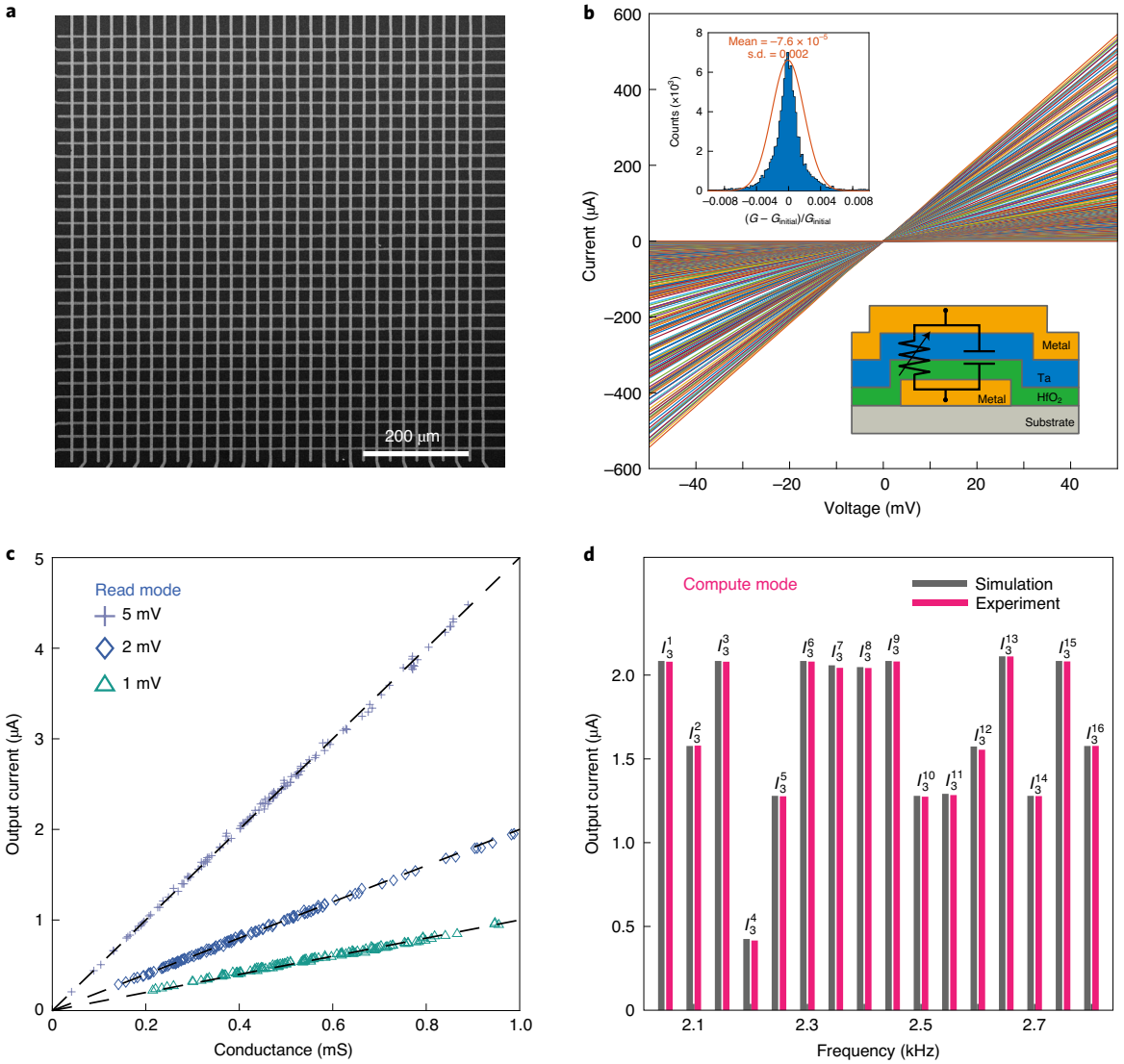


Fig. 2 | Experimental implementations of FMC-based massively parallel computing. a, Fabricated Ta/HfO_2 memristor crossbar array. **b**, Linear and symmetric *I*–*V* curves of the memristive devices at different conductance states. The insets show error distribution of device conductance (top) and schematic illustration of device structure (bottom). $G_{initial}$, initial conductance of memristors; *G*, measured conductance of memristors. **c**, Measured output currents versus conductance of different devices in the memristive crossbar array for $U_0 = 1$, 2 and 5 mV. For the same U_0, all the measured current values are located on a straight line with the slope equal to U_0, indicating that the parallel reading operation is valid. **d**, Comparison between experimental (grey histogram) and simulation (magenta histogram) current values at 16 frequencies in the massively parallel computing mode. The index labelled in each column corresponds to the output current at different frequencies.

the output current–frequency spectrum and read out the conductance values by using $G_{Read} = I_{Read}/U_0$, where I_{Read} is the measured current at different frequencies, and G_{Read} represents the conductance matrix. The measured output current values (symbols) at different U_0 values versus preprobed device conductance are presented in Fig. 2c. For the same U_0 (1, 2 or 5 mV), all measured current values are located in a straight line (dashed lines) with a slope equal to U_0, indicating that the accurate parallel reading is accessible in the memristive crossbar array.

Sequentially, we went a step further and realized massively parallel computing. By feeding continuous-time voltage signals with 16 frequency components into all of the rows of a 25×9 memristive crossbar array simultaneously, the corresponding continuous-time current output signals are generated instantly from all of the columns of the crossbar array. We analysed the current magnitudes at 16 frequencies generated from each column in the frequency domain, with the corresponding results shown in Supplementary Fig. 1. For simplicity, we selected the experimental current values at 16 frequencies output from the third column of the crossbar array (I_3^k) and compared them with the simulation results. Figure 2d shows a comparison between the experimentally measured current values (grey histogram) and the simulation current values (magenta histogram) at 16 frequency components (details for the experimental measurement and simulation in the Methods). As expected, the experimental results agree well with the simulation results. We also performed systematic analysis on error statistics over random inputs (Supplementary Fig. 2), as well as on the effects of wire resistance, crossbar array size and device resistance range (Supplementary Fig. 3), with results indicating that high-precision FMC-based parallel computing can be obtained. Implementing the one-shot MMM operations in the memristive crossbar array will achieve the massively parallel computing of numerous tasks and allow the real-time

inference that is desirable for many intelligent edge applications in the IoT network. Note that the massively parallel computing proposed in this work is radically different from parallel computing based on a multicore digital processor (Supplementary Fig. 4), in which parallel computing is limited by inherently sequential computing elements and the communication bottleneck still limits the computing performance[8].

By taking advantage of FMC-based parallel reading and computing, as demonstrated above, we are capable of achieving the one-shot recognition of numerous target images by using two memristive crossbar arrays. As shown in Fig. 3a, these two memristive crossbar arrays are physically interconnected by trans-impedance amplifiers (TIAs). The left crossbar array in this prototype is used to store the target letter images shown in Fig. 3b, and the right crossbar array is used as an artificial neural network for inference. To demonstrate the one-shot recognition of numerous target images, we mapped 16 letter images 'NAINVINAJNGINUHC' corresponding to a 25×16 data matrix (Fig. 3b) into the left crossbar array, and the trained weight matrix (Fig. 3c) into the right crossbar array. Subsequently, 16 different carrier signals, which have the same voltage amplitude U_0 but different frequencies (that is, from f_1 to f_{16}), were simultaneously fed to all columns of the left crossbar array to carry out parallel reading (in FMC-based Read mode). The output current signals from the left crossbar array that represent the 16 target letters are converted into continuous-time voltage signals and then input into the right crossbar array for classifying the target letters (in FMC-based Compute mode). The recognition results are output from the right crossbar array. With this unique set-up, the 16 letter images can be classified into nine different categories in a massively parallel one-shot manner, which are labelled 'A', 'N', 'J', 'I', 'G', 'V', 'U', 'C' and 'H'. To further show the potential of FMC technology in processing a large number of tasks, we input 1,000 hand-written digital letters for recognition. The obtained results show a recognition accuracy of 94.7% and indicate that our proposed FMC technology works well for massively parallel recognition of images (Supplementary Fig. 5).

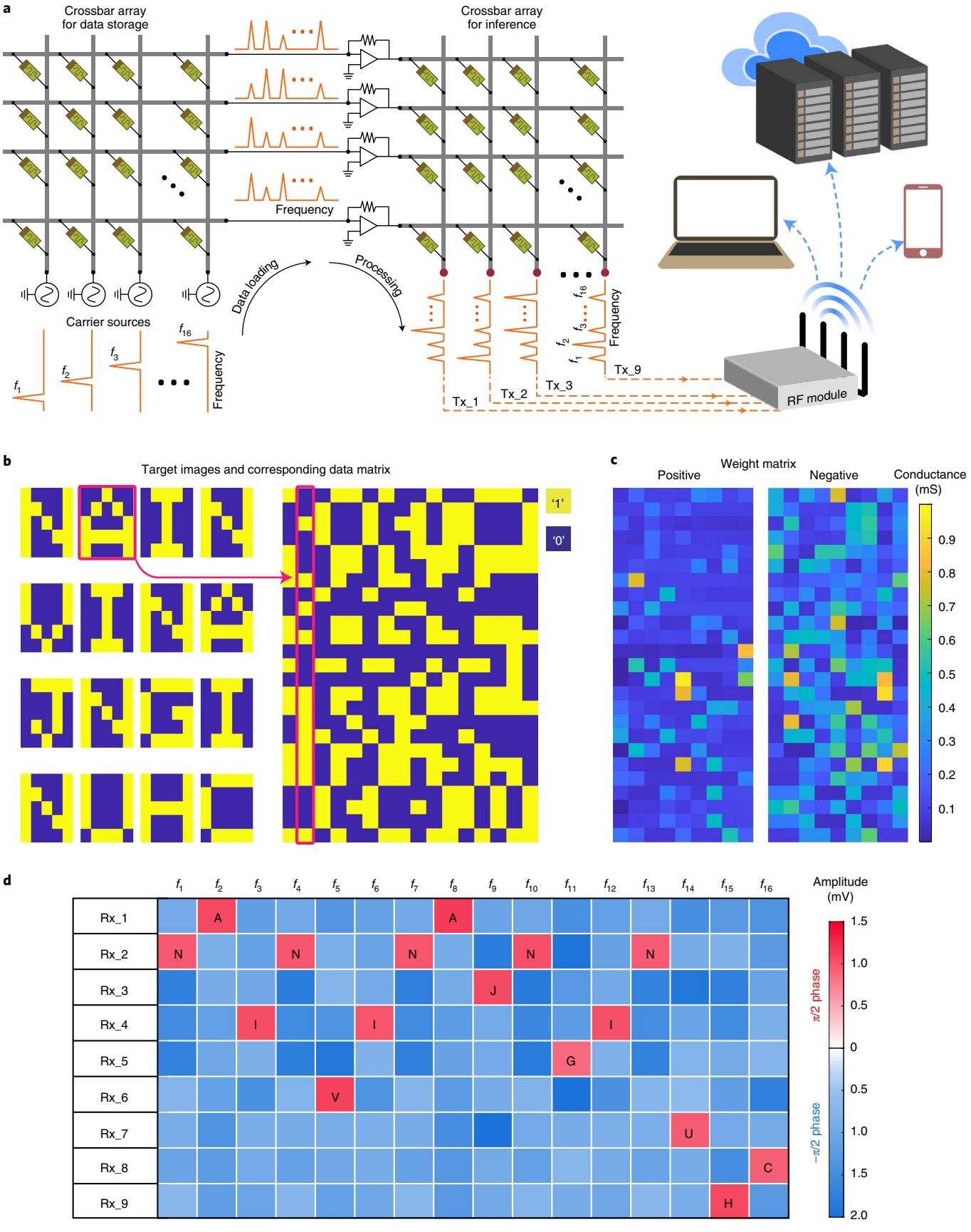
Note that the processed multiplexed signals output (representing recognized results) from all-analogue tiling of crossbar arrays can be either distinguished by exploiting highly parallelized processing in the end-to-end all-analogue crossbar architecture (Supplementary Fig. 6) or transmitted out by using a radio frequency (RF) module, depending on specific applications. Since the output multiplexed signals have been modulated, we can directly transmit recognized results through multiple-input-multiple-output (MIMO) wireless channels (Tx_1, Tx_2…Tx_9) without compromising performance. The transmitted signals are received on a remote terminal device over different wireless channels (Rx_1, Rx_2…Rx_9), as shown in Fig. 3d, in which the red boxes represent the classified target letters. We also experimentally demonstrated the reception of the recognition results on a remote mobile phone (Supplementary Fig. 7). With FMC-based technology, we are able to achieve massively parallel one-shot recognition of numerous target images, as well as signal modulation, transmission and reception in real time (Supplementary Video 1). Such transformative in-communication computing technology is desirable for advancing intelligent edge devices in IoT networks that demand high-efficiency data processing and communication. Moreover, we demonstrate that the FMC-based system is compatible with the MIMO communication technology widely used for increasing the channel capacity of 5G wireless communication networks[40] (Supplementary Fig. 8).

The use of the continuous-time data representation in the nanoscale crossbar array offers tremendous promise for reducing the operating voltage and increasing the computing frequency. Since the voltage U_0 of the carrier signal input into the left crossbar array in Fig. 3a is critical to the accuracy of the FMC-based massively parallel computing, we evaluated the parallel reading errors by using $(G_r - G_R)/G_r$ at different U_0 values, that is, from 1 mV to 100 mV (Fig. 4a and Supplementary Fig. 9), where G_r represents the readout conductance of the memristive crossbar devices by the semiconductor parameter analyser and G_R represents the conductance values obtained by the FMC. Our results show that the relative error is less than 2% at $U_0 = 2$ mV and is comparable to that reported in neuromorphic computing[37,39,41]. In contrast with the small error at low operating voltage, high operating voltage induced harmonic distortion, and conductance variation would lead to a large error (Supplementary Fig. 10). Note that a low-precision computation is sufficient for most neural network applications[29,36]. For similar-level precision, it is highly desirable to make the neural networks operate at a low voltage to achieve high energy efficiency. The operating voltage of the neural network based on the continuous-time data representation is two orders of magnitude lower than that of the digital circuit-based neural network[42–44] and reported values for the neural networks implemented with memristive crossbar arrays[35,38,41,45–47]. We reveal that the reason why massively parallel computing can be operated at ultralow voltage in the nanoscale crossbar array is due to highly suppressed noise at a high operating frequency (Fig. 4b). As the frequency increases, the signal-to-noise ratio is improved (inset of Fig. 4b). Although the proof-of-concept is demonstrated at a frequency of a few kilohertz, the operating frequency can be further increased. To explore the upper limit of the operating frequency, we carried out scattering-parameter measurements for the memristive devices ($0.4 \times 0.4\,\mu m^2$), with the results shown in Fig. 4c. The experimental S21 measurement matches well with the simulation model based on the equivalent circuit of the memristive devices (Supplementary Fig. 11), from which the capacitance can be extracted. Based on the extracted capacitance (~20 fF), we obtained an operating frequency of 5 GHz even for a 128×128 crossbar array made of the large-area memristive device (Methods for more details). By further scaling down the feature size of the memristive device to reduce the parasitic capacitance[48–50], it is possible to engineer the operating frequency of the FMC-based massively parallel computing in memristor-based MMM cores beyond 100 GHz (Fig. 4d). The full matrix multiplication in a single time step is available within this bandwidth limit and reasonable noise level. Such a wide operating frequency range enables one to add more frequency components in the input continuous-time signal to increase the parallel computing capability. In conjunction with recent advances in the integration density of memristive crossbar arrays[33,38], the parallel processing capability can be further enhanced by increasing the array size and the number of tiled arrays (Supplementary Fig. 12).

In conclusion, we propose and experimentally demonstrate a frequency-multiplexing-enabled massively parallel computing scheme based on a continuous-time data representation in the nanoscale crossbar array. The ultralow operating voltage and ultrahigh operating frequency of the FMC-based massively parallel computing may open up an avenue for parallel analogue computers

Fig. 3 | FMC-based one-shot recognition of numerous images and wireless communication of the recognition results. a, The circuit schematic of the device based on two crossbar arrays and a RF module. The left and right crossbar arrays are used for data storage and inference, respectively. **b**, The data matrix for 16 target images 'NAINVINAJNGINUHC' stored in the left crossbar array. **c**, The trained weight matrix for the right crossbar array. **d**, One-shot recognition results of the 16 letter images. The recognition results were transmitted via the RF module and received by the terminal device through wireless communications. The red boxes in each row (Rx_1 to Rx_9) correspond to the classified target letters from the 16 input letters represented by different frequency components ($f_1, f_2 \cdots f_{16}$).

that are superior to their digital counterparts with orders of magnitude improvement in the voltage and computing speed. In conjunction with the signal modulation accomplished simultaneously with parallel signal processing, FMC-based massively parallel computing may be deployed for low-power intelligent edge applications to address the upcoming challenges associated with real-time



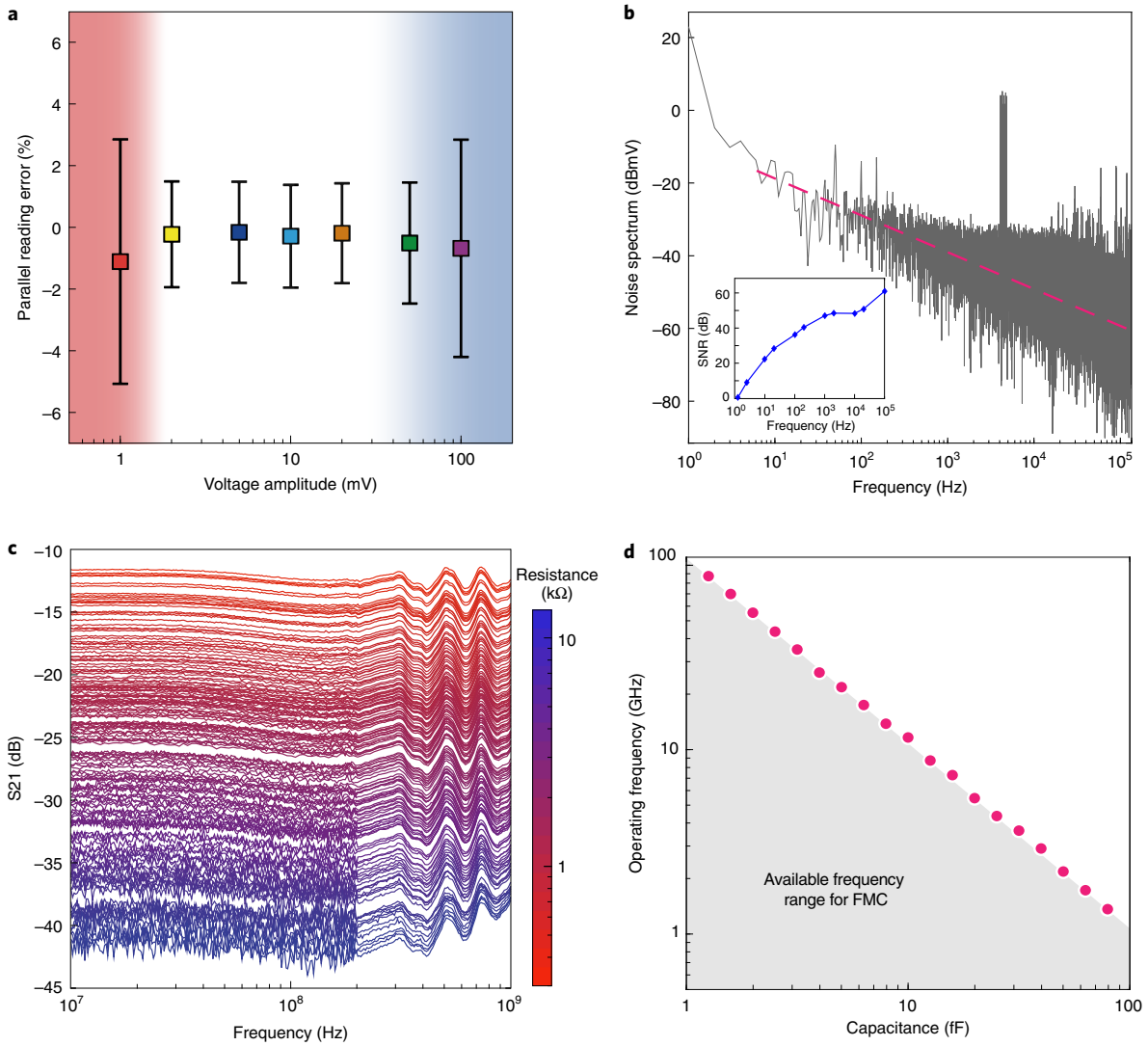


Fig. 4 | Performance of FMC-based massively parallel computing. a, The relative error is evaluated at various voltage amplitudes (U_0) of the carrier signals. The red and blue shading is used to indicate that the relatively large errors are caused by the low SNR and the nonlinearity, respectively. **b**, Noise spectrum measured for the memristive device. The dashed red line is used to indicate the trend of noise suppression with increasing operating frequency. The inset shows that the SNR increases with the frequency. **c**, Scattering-parameter S21 measurement of the memristive device by using a vector network analyser at different conductance levels (solid lines). **d**, The operating frequency of the FMC-based massively parallel computing is calculated based on the capacitance of a nanoscale memristive device. Note that similar results can be obtained in memristors with higher resistance values (Supplementary Fig. 13). SNR, signal-to-noise ratio.

processing and communication in IoT networks[51]. The proposed FMC could be readily extended to other nanoscale crossbar arrays made of non-volatile memory devices[13,32,52–57] (for example, phase change memory and magnetoresistive random access memory).

Note that while our manuscript was under review, we noticed two recent publications presenting photonic tensor cores for parallel computing[58,59] and made a detailed comparison between the optical parallel computing and our proposed FMC (Supplementary Tables 1 and 2).

Online content

Any methods, additional references, Nature Research reporting summaries, source data, extended data, supplementary information, acknowledgements, peer review information; details of author contributions and competing interests; and statements of data and code availability are available at https://doi.org/10.1038/s41565-021-00943-y.

Received: 19 November 2020; Accepted: 11 June 2021;
Published online: 8 July 2021

References

1. Markov, I. L. Limits on fundamental limits to computation. *Nature* **512**, 147–154 (2014).
2. Zhirnov, V. V., Cavin, R. K., Hutchby, J. A. & Bourianoff, G. I. Limits to binary logic switch scaling — a gedanken model. *Proc. IEEE* **91**, 1934–1939 (2003).
3. Waldrop, M. M. The chips are down for Moore's law. *Nature* **530**, 144–147 (2016).
4. Yasumoto, K., Yamaguchi, H. & Shigeno, H. Survey of real-time processing technologies of IoT data streams. *J. Inf. Process.* **24**, 195–202 (2016).
5. Di Ventra, M. & Pershin, Y. V. The parallel approach. *Nat. Phys.* **9**, 200–202 (2013).
6. El-Kareh, B. & Hutter, L. N. *Silicon Analog Components* (Springer, 2015).
7. Big data needs a hardware revolution. *Nature* **554**, 145–146 (2018).
8. Végh, J. How Amdahl's Law limits the performance of large artificial neural networks. *Brain Inform.* **6**, 4 (2019).

9. Krestinskaya, O., James, A. P. & Chua, L. O. Neuromemristive circuits for edge computing: a review. *IEEE Trans. Neural Netw. Learn. Syst.* **31**, 4–23 (2020).
10. Yang, Y. Multi-tier computing networks for intelligent IoT. *Nat. Electron.* **2**, 4–5 (2019).
11. Cai, F. et al. Power-efficient combinatorial optimization using intrinsic noise in memristor Hopfield neural networks. *Nat. Electron.* **3**, 409–418 (2020).
12. Liu, C. et al. Small footprint transistor architecture for photoswitching logic and in situ memory. *Nat. Nanotechnol.* **14**, 662–667 (2019).
13. Marković, D., Mizrahi, A., Querlioz, D. & Grollier, J. Physics for neuromorphic computing. *Nat. Rev. Phys.* **2**, 499–510 (2020).
14. Miscuglio, M. & Sorger, V. J. Photonic tensor cores for machine learning. *Appl. Phys. Rev.* **7**, 031404 (2020).
15. Lin, X. et al. All-optical machine learning using diffractive deep neural networks. *Science* **361**, 1004–1008 (2018).
16. Feldmann, J., Youngblood, N., Wright, C. D., Bhaskaran, H. & Pernice, W. H. P. All-optical spiking neurosynaptic networks with self-learning capabilities. *Nature* **569**, 208–214 (2019).
17. Kumar, S., Williams, R. S. & Wang, Z. Third-order nanocircuit elements for neuromorphic engineering. *Nature* **585**, 518–523 (2020).
18. Sebastian, A., Le Gallo, M., Khaddam-Aljameh, R. & Eleftheriou, E. Memory devices and applications for in-memory computing. *Nat. Nanotechnol.* **15**, 529–544 (2020).
19. Bandyopadhyay, A., Pati, R., Sahu, S., Peper, F. & Fujita, D. Massively parallel computing on an organic molecular layer. *Nat. Phys.* **6**, 369–375 (2010).
20. Chai, Y. In-sensor computing for machine vision. *Nature* **579**, 32–33 (2020).
21. Wang, C.-Y. et al. Gate-tunable van der Waals heterostructure for reconfigurable neural network vision sensor. *Sci. Adv.* **6**, eaba6173 (2020).
22. Pan, C. et al. Reconfigurable logic and neuromorphic circuits based on electrically tunable two-dimensional homojunctions. *Nat. Electron.* **3**, 383–390 (2020).
23. Wang, S. et al. Networking retinomorphic sensor with memristive crossbar for brain-inspired visual perception. *Natl Sci. Rev.* **8**, nwaa172 (2020).
24. Shannon, C. E. Mathematical theory of the differential analyzer. *J. Math. Phys.* **20**, 337–354 (1941).
25. Zhang, W. et al. Neuro-inspired computing chips. *Nat. Electron.* **3**, 371–382 (2020).
26. Ielmini, D. & Wong, H. S. P. In-memory computing with resistive switching devices. *Nat. Electron.* **1**, 333–343 (2018).
27. Sun, Z. et al. Solving matrix equations in one step with cross-point resistive arrays. *Proc. Natl Acad. Sci. USA* **116**, 4123–4128 (2019).
28. Zidan, M. A. et al. A general memristor-based partial differential equation solver. *Nat. Electron.* **1**, 411–420 (2018).
29. Xia, Q. & Yang, J. J. Memristive crossbar arrays for brain-inspired computing. *Nat. Mater.* **18**, 309–323 (2019).
30. Zidan, M. A., Strachan, J. P. & Lu, W. D. The future of electronics based on memristive systems. *Nat. Electron.* **1**, 22–29 (2018).
31. Pi, S. et al. Memristor crossbar arrays with 6-nm half-pitch and 2-nm critical dimension. *Nat. Nanotechnol.* **14**, 35–39 (2019).
32. Sebastian, A., Le Gallo, M. & Eleftheriou, E. Computational phase-change memory: beyond von Neumann computing. *J. Phys. D* **52**, 443002 (2019).
33. Chen, W.-H. et al. CMOS-integrated memristive non-volatile computing-in-memory for AI edge processors. *Nat. Electron.* **2**, 420–428 (2019).
34. Wang, M. et al. Robust memristors based on layered two-dimensional materials. *Nat. Electron.* **1**, 130–136 (2018).
35. Yao, P. et al. Fully hardware-implemented memristor convolutional neural network. *Nature* **577**, 641–646 (2020).
36. Prezioso, M. et al. Training and operation of an integrated neuromorphic network based on metal-oxide memristors. *Nature* **521**, 61–64 (2015).
37. Sheridan, P. M. et al. Sparse coding with memristor networks. *Nat. Nanotechnol.* **12**, 784–789 (2017).
38. Lin, P. et al. Three-dimensional memristor circuits as complex neural networks. *Nat. Electron.* **3**, 225–232 (2020).
39. Yao, P. et al. Face classification using electronic synapses. *Nat. Commun.* **8**, 15199 (2017).
40. Raleigh, G. G. & Cioffi, J. M. Spatio-temporal coding for wireless communication. *IEEE Trans. Commun.* **46**, 357–366 (1998).
41. Hu, M. et al. Memristor-based analog computation and neural network classification with a dot product engine. *Adv. Mater.* **30**, 1705914 (2018).
42. *International Roadmap for Devices and Systems: More Moore* 2017 edn (IEEE, 2018).
43. Merolla, P. A. et al. A million spiking-neuron integrated circuit with a scalable communication network and interface. *Science* **345**, 668 (2014).
44. Davies, M. et al. Loihi: a neuromorphic manycore processor with on-chip learning. *IEEE Micro* **38**, 82–99 (2018).
45. Yeon, H. et al. Alloying conducting channels for reliable neuromorphic computing. *Nat. Nanotechnol.* **15**, 574–579 (2020).
46. Choi, S. et al. SiGe epitaxial memory for neuromorphic computing with reproducible high performance based on engineered dislocations. *Nat. Mater.* **17**, 335–340 (2018).
47. Cai, F. et al. A fully integrated reprogrammable memristor–CMOS system for efficient multiply–accumulate operations. *Nat. Electron.* **2**, 290–299 (2019).
48. Pi, S. et al. Nanoscale memristive radiofrequency switches. *Nat. Commun.* **6**, 7519 (2015).
49. Torrezan, A. C. et al. Sub-nanosecond switching of a tantalum oxide memristor. *Nanotechnology* **22**, 485203 (2011).
50. Kim, M. et al. Analogue switches made from boron nitride monolayers for application in 5G and terahertz communication systems. *Nat. Electron.* **3**, 479–485 (2020).
51. Satyanarayanan, M. How we created edge computing. *Nat. Electron.* **2**, 42 (2019).
52. Fuller, E. J. et al. Parallel programming of an ionic floating-gate memory array for scalable neuromorphic computing. *Science* **364**, 570–574 (2019).
53. Sebastian, A. et al. Tutorial: brain-inspired computing using phase-change memory devices. *J. Appl. Phys.* **124**, 111101 (2018).
54. Burr, G. W. et al. Experimental demonstration and tolerancing of a large-scale neural network (165 000 synapses) using phase-change memory as the synaptic weight element. *IEEE Trans. Electron Devices* **62**, 3498–3507 (2015).
55. Wong, H. S. P. & Salahuddin, S. Memory leads the way to better computing. *Nat. Nanotechnol.* **10**, 191–194 (2015).
56. Arimoto, Y. & Ishiwara, H. Current status of ferroelectric random-access memory. *MRS Bull.* **29**, 823–828 (2004).
57. Zhang, W., Mazzarello, R., Wuttig, M. & Ma, E. Designing crystallization in phase-change materials for universal memory and neuro-inspired computing. *Nat. Rev. Mater.* **4**, 150–168 (2019).
58. Feldmann, J. et al. Parallel convolutional processing using an integrated photonic tensor core. *Nature* **589**, 52–58 (2021).
59. Miscuglio, M. & Sorger, V. J. Photonic tensor cores for machine learning. *Appl. Phys. Rev.* **7**, 031404 (2020).

Methods

Device fabrication. Pd/Ta/HfO_2/Pd memristive devices were fabricated with a sandwich structure. An Al_2O_3 substrate was used to eliminate the parasitic capacitance. The bottom/top metal layer was deposited through a standard electron beam deposition process; a 6-nm-thick HfO_2 switching layer was deposited via atomic layer deposition. The deposition of an 80-nm-thick Ta layer was realized by using a standard RF sputtering process. For devices with a feature line-width larger than 2 μm, the electrode patterns were realized by using double-layer photoresist photolithography, followed by a lift-off process in *N*-methyl pyrrolidone. For line-widths smaller than 2 μm, electron beam lithography was used to pattern the electrodes.

Implementation of massively parallel computing in memristive crossbar array. Memristor resistance values ranging from 1,000 to 10,000 ohm were used for the demonstration. To demonstrate massively parallel computing in the memristive crossbar array, signal generators, TIAs and an oscilloscope with a built-in frequency analyser were used. Signal generators supporting 16 channels were used as continuous-time signal sources. We employed TIAs to convert the current signals into voltage signals for the measurement; a frequency analyser was used to measure the output results from the memristive crossbar array. The output signals can also be distinguished in a massively parallel way with the crossbar array (Supplementary Fig. 6).

Implementation of parallel reading in the memristive crossbar array. To demonstrate the parallel reading of the data stored in the crossbar array, we used the individual-frequency sinusoidal voltage signals as carrier signals and applied them into different rows of the crossbar array, where the conductance matrix is written randomly. The output current was converted into a voltage by the TIAs and was subsequently measured by the frequency analyser. In Read mode, multiple current peaks are present in the current–frequency spectrum output from each column of the crossbar array. Based on the specific row in which a carrier signal with distinct frequency was applied, the frequency corresponding to a specific current peak can be distinguished at each column of the crossbar array, in which the conductance of the selected memristor is proportional to the specific current magnitude (or peak).

Measurement of the S21 parameter on the memristive device. We used a Tektronix TTR506A vector network analyser to apply high-frequency microwave signals to the memristive devices to measure the scattering parameter. The S21 parameter represents the forward transmission gain through the memristive devices. The S21 curves in Fig. 4c were measured on a memristive device with an area of 0.16 μm^2. Different S21 curves were obtained at different conductance levels (ranging from 300 to 13,000 Ω). A simulation model with a variable resistor connected in parallel with a constant capacitor was developed to calculate S21. The simulation results are in good agreement with the experimental results (Supplementary Fig. 11). By fitting to the measured S21 curves with the model, we extracted the capacitance of the 0.16 μm^2 memristor to be 20 fF.

Operating frequency of FMC-based massively parallel computing scheme. At high frequencies, the accuracy of FMC is mainly limited by the parasitic effect of the memristors, considering that the effects of the wire resistance and capacitance associated with the adjacent wires are negligible (Supplementary Fig. 14). Scaling down the feature size of the memristive devices can significantly reduce the parasitic capacitance and increase the cut-off frequency of FMC (Supplementary Fig. 15). Such a parasitic effect can also be mitigated by using the differential scheme (Supplementary Fig. 16). To obtain the results shown in Fig. 4d, we used a 128 × 128 memristive crossbar array, with memristor resistance varying from 1 to 10 kΩ and parasitic capacitances of the memristors obeying a Gaussian distribution with a relative standard deviation of 5%. The cut-off error of FMC was fixed at 1%.

Training of the artificial neural network. To reduce the power consumption in the memristor arrays and mitigate the effect of the wire resistance, high-resistance states were preferred when programming the memristive crossbar array. A weight change was realized by stimulating an alternative memristor in a differential pair, $W = G_+ - G_-$, as the weight (W) is proportional to the difference between the conductance matrices of positive crossbar array (G_+) and negative crossbar array (G_-). All memristors in both positive and negative crossbar arrays were first set into high resistive states. Based on the gradient descent algorithm, an online training process was performed. Subsequently, the expected conductance was mapped into a memristive crossbar array. If the weight change $\Delta G(i,j) > 0$, then a positive pulse would be applied to increase G_+, and a negative pulse would be used to decrease G_-; otherwise, a positive pulse would be applied to increase G_-, and a negative pulse would be used to decrease G_+.

Channel capacity of the communication system. The output signals of FMC could be transmitted by antennas in the analogue domain, with full compatibility with a MIMO/orthogonal-frequency-division-multiplexing (MIMO-OFDM) wireless communication system for IoT applications. A wide-band MIMO-OFDM communication system with N_t transmitting antennas and N_r receiving antennas was considered. The MIMO system model at the kth subcarrier is given as

$$\mathbf{y}_k = H_k \mathbf{s}_k + \mathbf{n}_k, \tag{1}$$

where $\mathbf{y}_k \in \mathbb{C}^{N_r}$ and $\mathbf{s}_k \in \mathbf{\Omega}^{N_t}$ are the received and transmitted vectors at the kth subcarrier, respectively, and the constellation of each component s_i is denoted by $\mathbf{\Omega}$. $H_k \in \mathbb{C}^{N_r \times N_t}$ is the channel matrix at the kth subcarrier and assumed to be perfectly known at the receiving terminal, and $\mathbf{n}_k \sim \mathcal{CN}\left(0,\ \sigma_k^2 I_{N_r}\right)$ is the additive white Gaussian noise vector at the kth subcarrier, in which σ_k is the standard deviation and I_{N_r} is an identity matrix of $N_r \times N_r$. By applying information theory to the MIMO-OFDM system model, the capacity can be obtained as follows:

$$C = \sum_k B_k \max_{S_k : tr(S_k) \le P_k} \log_2 \left| I_{N_r} + \frac{1}{\sigma_k^2} H_k S_k H_k^H \right|, \tag{2}$$

where C is the channel capacity of the MIMO-OFDM system, B_k and P_k are the bandwidth and transmitting power of the kth subcarrier, respectively, H_k^H is the conjugate transpose of the matrix H_k, and S_k denotes the covariance matrix of the transmitting signal vector at the kth subcarrier. Note that the bandwidth of each subcarrier is set to be the same in a realistic OFDM system. The average channel capacity with respect to the number of transmitting/receiving antennas can be evaluated for different numbers of subcarriers. Supplementary Fig. 8 shows the simulation results, in which an independent and identically distributed Rayleigh-fading channel matrix was used. As shown in the figure, the channel capacity of the MIMO-OFDM system is approximately proportional to the number of transmitting/receiving antennas at a large-scale antenna array. In addition, the channel capacity can also be enhanced by increasing the number of subcarriers. These results indicate that the transmission rate of the recognized results output from the FMC system can be further increased by adopting MIMO technology and increasing the number of subcarriers.

Evaluation of computing capability of the FMC system. The hardware performance of the FMC system can be evaluated by fully performing in-memory multiply–accumulate (MAC) operations. To calculate the MAC operations, we have utilized the method adopted for photonic wave division multiplexing[58]. With the available memristive crossbar array (32 × 32) and 16 multiplexed frequencies, the MAC operations per second of the FMC system can be calculated as follows: MAC operations = multiplexed frequencies (16) × crossbar array size (32 × 32) × modulating speed (1 GHz) = 16 tera-operations per second (TOPS). Note that the number of multiplexed frequencies is practically limited by the number of columns of the memristive crossbar array, which can be readily improved by increasing the array size. Since the modulation of the input signal is implemented by the crossbars and complementary metal–oxide–semiconductor circuits (Supplementary Fig. 17), its modulating speed is determined by the gate delay of the transistors, which can be shorter than 1 ns. Therefore, it is reasonable to use the upper bound of gate delay as the modulating speed (1 GHz). Increasing the array size can implement more MAC operations in a single time step and enhance the computing capability of the FMC.

To compare the FMC technology with alternative technologies, it is desirable to calculate the computing density and computing efficiency. As a prototype demonstration, the peripheral circuit was not integrated with crossbar arrays. To analyse the area cost and energy cost associated with the peripheral circuitry, we employed electronic design automation tools, widely used in the industry for designing and verifying integrated circuits to design the peripheral circuitry. Based on a 65 nm complementary metal–oxide–semiconductor technology node, we calculated the power consumption and the area of individual components included in the peripheral circuitry, with the corresponding results shown in Supplementary Table 3. The system-level peripheral circuitry used for MAC operations is given in Supplementary Fig. 18. Based on these parameters, the computing density of the FMC is estimated to be 10.7 TOPS mm^{-2} (MAC operations per second/(array area + peripheral circuit area) = 16 TOPS per 1.5 mm^2). The computing efficiency is estimated to be 9.3 TOPS W^{-1} ((MAC operations per second)/(total power consumption) = 16 TOPS per 1.72 W).

Data availability

The data supporting the findings of this study are available within the article and its Supplementary Information, and from the corresponding author upon reasonable request.

Acknowledgements

This work was supported in part by the National Natural Science Foundation of China (62034004, 61625402, 61974176 and 61921005), the Strategic Priority Research Program of the Chinese Academy of Sciences (XDB44000000), the National Key R&D Program of China (2019YFB2205400 and 2019YFB2205402) and Fundamental Research Funds for the Central Universities (020414380179 and 020414380171). F.M. acknowledges the support from the AIQ foundation and experimental assistance from Q. Liu, X. Tan and Z. Wu.

Author contributions

F.M., S.-J.L. and C.W. conceived the idea and designed the experiments. F.M. and S.-J.L. supervised the whole project. C.W. performed all experiments. C.W. and S.-J.L. analysed

the experimental data. C.-Y.W. and C.P. provided assistance during the experiment design. Z.-Z.Y. assisted in the device fabrication and circuit assembly. X.S. and W.W. contributed to circuit measurement. Y.G., Z.Z. and C.Z. contributed to the MIMO model. C.W. and Y.Z. carried out the simulation of the circuit models. C.W., S.-J.L. and F.M. co-wrote the manuscript.

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/s41565-021-00943-y.

Correspondence and requests for materials should be addressed to F.M.

Peer review information *Nature Nanotechnology* thanks Yang Chai, Suhas Kumar and Abu Sebastian for their contribution to the peer review of this work.

Reprints and permissions information is available at www.nature.com/reprints.