SuperMind: a survey of the potential of superconducting electronics for neuromorphic computing

OPEN ACCESS
IOP Publishing
Superconductor Science and Technology
Supercond. Sci. Technol. **35** (2022) 053001 (26pp)
https://doi.org/10.1088/1361-6668/ac4cd2

Topical Review

SuperMind: a survey of the potential of superconducting electronics for neuromorphic computing

Michael Schneider[1,*], **Emily Toomey**[2], **Graham Rowlands**[3], **Jeff Shainline**[1], **Paul Tschirhart**[4] **and Ken Segall**[5]

[1] National Institute of Standards and Technology, Boulder, CO 80305, United States of America
[2] Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, Cambridge, MA 02138, United States of America
[3] Quantum Engineering and Computing, Raytheon BBN Technologies, Cambridge, MA 02138, United States of America
[4] Advanced Technology Laboratory, Northrop Grumman Corporation, Linthicum, MD 21090, United States of America
[5] Department of Physics and Astronomy, Colgate University, Hamilton, NY 13346, United States of America

E-mail: michael.schneider@nist.gov

Received 21 June 2021, revised 12 October 2021
Accepted for publication 18 January 2022
Published 30 March 2022

Abstract

Neuromorphic computing is a broad field that uses biological inspiration to address computing design. It is being pursued in many hardware technologies, both novel and conventional. We discuss the use of superconductive electronics for neuromorphic computing and why they are a compelling technology for the design of neuromorphic computing systems. One example is the natural spiking behavior of Josephson junctions and the ability to transmit short voltage spikes without the resistive capacitive time constants that typically hinder spike-based computing. We review the work that has been done on biologically inspired superconductive devices, circuits, and architectures and discuss the scaling potential of these demonstrations.

Keywords: neuromorphic, Josephson junction, bio-inspired, neural network

(Some figures may appear in colour only in the online journal)

1. Introduction

The human brain offers some insight into a method of computation that differs from typical digital logic. The brain is a highly parallel computing platform that exhibits tremendous adaptability and fault tolerance while simultaneously being extremely energy efficient. Neuromorphic computing seeks to take advantage of several of these characteristics by implementing biologically inspired design directly in hardware. The potential power of biologically inspired computing is evidenced in the growing prevalence of artificial intelligence or machine learning algorithms. Because of the success of these algorithms, the computational need for training them has been doubling every four months, nearly five times the exponential growth rate offered by Moore's Law operating at its peak.

* Author to whom any correspondence should be addressed.

However, these algorithms are still predominantly trained and run on hardware that is optimized for digital logic with its associated memory bottleneck and high power consumption. It should be noted that recent work on digital superconducting architectures has shown that significant improvements can be made in the design of superconducting systems that can train and run software based neural networks [1]. These systems compare favorably in both energy and speed compared with purpose built CMOS hardware designed for the same task [2]. While such advances in digital superconducting systems are very important, in this review we survey the analog and mixed analog–digital use of biologically inspired computing design from the device up, and its potential to build an efficient neuromorphic computational stack.

Josephson junctions (JJs) are naturally neuromorphic hardware devices that can implement several biological primitives at the device level. Examples of these primitives include the spiking nature of neurons in the human brain, which can be mimicked with a single JJ. More biologically realistic spiking behavior including following the basic ion flow dynamics of Na and Ca within a neuron can be implemented with a two JJ circuit [3]. The near lossless transmission of voltage spikes that occur in the human brain can be easily implemented with active or passive superconducting transmission lines [4]. The memory and plasticity mechanisms in the brain, which are implemented with modulations of the synaptic strength between neurons, can also be implemented with a simple two-JJ circuit or a novel magnetic JJ, among other options [5, 6]. We discuss architectures that explicitly set the weights of these artificial synapses as well as those that make use of hidden-layer dynamics whose internal weights are less important than their overall topology.

The number of neurons in the brain is on the order of 10^{11} and the number of synapses is on the order of 10^{14} [7]. These numbers exceed the largest JJ circuits, which are on the order of 10^{6} [8]. However, many potential applications do not require such a large junction count and therefore seem tractable with the current state of JJ fabrication technology. Further, one particularly compelling trait of neuromorphic JJ circuits is the potential to operate at tens or hundreds of gigahertz achieving many orders of magnitude in speed over the human brain, which typically operates below 1 kHz. In addition, the energy consumption of spiking JJs can readily be below 10^{-18} J, also giving them an advantage compared to the energy needed to produce a spike in the human brain, which typically requires on the order of 10^{-14} J [7]. This energy advantage holds true even after taking into account the cooling requirements to operate at 4.2 K, which can be less than 1000 W to cool 1 W at 4.2 K [9]. Another critical departure of the human brain from typical modern computer architectures is its extremely high connectivity. Large-scale fan-out and fan-in can be achieved in single flux quantum architectures but they require a commensurate number of additional JJ elements. The way JJs are connected together in an architecture will most likely vary depending on the target application; we discuss some potential choices and their trade-offs below.

In the review that follows, we cover the current state of neuromorphic superconducting electronics. We organize the paper starting with devices and moving our way up the computational stack. Work in this field is at various states across this range. A brief summary of what has been accomplished so far is as follows. Experimentally there have been several exciting demonstrations. These include compact (2 JJ) circuits that exhibit biologically plausible neuron behavior, synapse circuits based on SQUIDs and synapse devices based on single magnetic JJs. There have also been experimental demonstrations of binary JJ synapses. In addition, the size, shape and probabilistic nature of the action potential was experimentally verified in Nanowire neurons. In addition, two JJ neurons coupled with a SQUID synapse have been experimentally demonstrated and show several biological behaviors, such as a phase-flip bifurcation. The field in general leverages the significant work in the more well established fields of JJs, superconducting nanowires, single photon detectors, and digital logic based on single flux quantum circuits and adiabatic quantum-flux-parametron circuits.

Detailed SPICE simulations have been made using compact models that have been well tested within the field of superconducting digital logic. These simulations include the fan-in and fan-out based on SFQ digital logic circuits but applied to highly connected neuromorphic circuits. Also, detailed SPICE simulations of the implementation of standard software-based neural networks directly in superconducting hardware, both JJ based and nanowire based have been performed. SPICE simulations have been used to show how JJs can make a dynamical system, or reservoir network, that can be used for signal recovery in high speed communications. Finally, SPICE simulations of a winner-take-all network based on superconducting nanowires were demonstrated.

Finally, there has been significant conceptual work and analysis looking further up the computational stack at how best to make neuromorphic architectures based on both superconducting hardware and also looking at the potential benefits of integrating photonics with superconducting hardware for large-scale connectivity. The architectures that have been considered included small blocks of all-to-all connectivity of analog neuromorphic JJ circuits, spiking neural networks, as well as mixed analog and digital JJ systems. The various trade offs that can be made in the implementation of each of these architectures has also been studied. In the case of mixed analog and digital systems this trade off analysis was based on results from experiments involving superconducting digital logic. The integration of optics with superconducting hardware is also theoretically studied for optoelectronic neurons and synapses, where they have good potential to implement biological functionality.

2. Building blocks

2.1. Neurons

A simplified model of a biological neuron consists of four basic components, as illustrated in figure 1. Dendrites receive

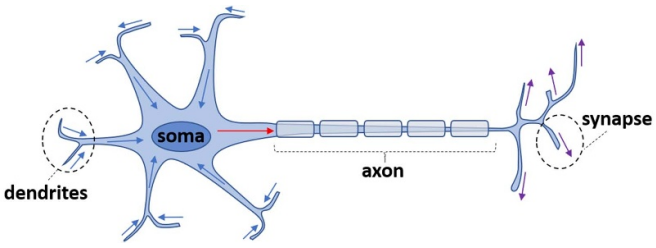

Figure 1. Basic model of a biological neuron, featuring the four primary building blocks: dendrites, the soma, the axon, and synapses.

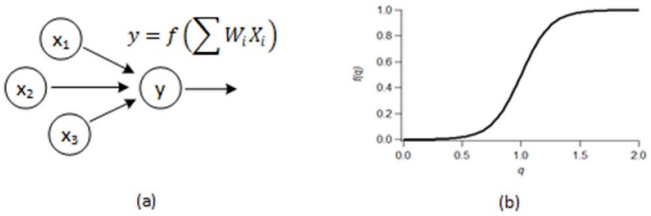

Figure 2. Neural network overview: (a) schematic of a simple neural network, with neurons as circles and arrows as weighted connections. The state of the output neuron Y depends on the weighted sum of the inputs X. (b) The function f(q) determines the threshold response of each neuron. (c) A JJ implementation of a neuron, proposed by Yamanashi *et al* [11], where the output probability of an SFQ pulse depends on the input current. (d) Simulated and measured response of the circuit from part (c).

and integrate input signals from the outputs of other connecting neurons and transmit them into the cell body, or soma, where they are summed. If the sum of the input signals exceeds a certain threshold, the soma generates an electrical spike known as an action potential. Action potentials are the key unit of information in brain communications. Once an action potential is created, it travels down the long extension of the cell body called the axon [10]. In addition to serving as a propagation pathway, axons can act like transmission lines, delaying action potentials so that their temporal information can be preserved. Most importantly, axons feed action potentials into synapses, which serve as the critical connection point between two neurons. In response to an action potential, synapses either suppress or excite the firing of a downstream neuron, often through the release of chemical neurotransmitters. As discussed in more detail in section 2.2, the strength of a synaptic connection is variable and becomes stronger the more often two neurons fire together. This mechanism forms the basis of learning in both biological and artificial spiking neural networks.

Neuromorphic computing, simply speaking, involves utilizing the building blocks of figure 1 to construct an information processing unit. Some approaches try to mimic the brain as closely as possible, utilizing features like the timing information of the spikes, the plasticity of the synapses, and the interplay of the ion channel dynamics in the soma, among others. Other approaches are more minimalist, utilizing only the stored weights and the nonlinearity of the threshold response. We discuss this simpler approach first, and then follow with those that are more biologically realistic.

The term 'Neural Network' refers to an interconnected assembly of simple processing units or nodes, where each node is a realization of the soma presented in figure 1. In the simplest case, these nodes are in one of two states, either 'on' or 'off' and outputting a 1 or 0, respectively. These nodes connect to other nodes through weighted connections. The signal flowing down each connection is the output of the node times the weight of the connection. These signals are then summed together in the downstream node, possibly altering its state. A simple schematic is shown in figure 2(a), where X_1, X_2 and X_3 represent three nodes which are connected to fourth node Y. The state of Y is determined by the sum ($W_1X_1 + W_2X_2 + W_3X_3$), where W_i represents the weight of connection i. Roughly speaking, if this sum is greater than the threshold of Y, then Y is 'on' and outputs a 1; if the sum is less than the threshold of Y then Y is 'off' and outputs a 0. The exact function describing the state of Y is given by a thresholding function $f(q)$, an example of which is shown in figure 2(b). Here f ($\sum W_iX_i$) is the output of the node. Sharper thresholding functions can form more powerful networks; however, more rounded functions like the one shown can ease training requirements.

Yamanashi *et al* [11] developed a comparator for the implementation of a sigmoid thresholding function. A current was the input, equivalent to q in figure 2(b), and the output state was indicated by the probability of releasing a single flux quantum (SFQ) pulse on a given clock cycle, equivalent to the function f. Figure 3 of their manuscript shows the electrical diagram of the comparator. If the input current caused the current of the comparator junction to exceed its critical current, then an SFQ pulse was released and the output was a 1; if this current failed to exceed the critical current, then the output was 0. Thermal activation can round this function, but in general the switching of a JJ is a discrete event, resulting in a sharp threshold and a high degree of nonlinearity. The actual experimental graph of output probability versus input current is shown in figure 4 of their manuscript, where the similarity to figure 2(b) is evident.

A single layer network like that shown in figure 2(a) is also called a perceptron, due to its ability to 'recognize' a certain input. For example, if we wanted the output Y to change state only when input X_1 was on, we could make weight W_1 large while keeping W_2 and W_3 small, such that the sum ($W_2X_2 + W_3X_3$) was below the threshold of Y whereas W_1X_1 was above the threshold of Y. Similarly, if we wanted Y to 'recognize' when X_1 and X_2 were on, we could adjust parameters such that only ($W_1X_1 + W_2X_2$) was above threshold, while any other combination was not. If we then drastically increase the number of nodes and the number of layers, the system can become capable of recognizing very complex data patterns. This ability has given neural networks a wide range of applications.

Implicit in this scheme is that the product of any state times a weight (e.g. W_iX_i) is represented physically by a current, since that is the x-axis for the thresholding function. To convert the outgoing SFQ pulse into a current, Yamanashi *et al* [11] use a clever scheme to convert the rate of SFQ pluses leaving

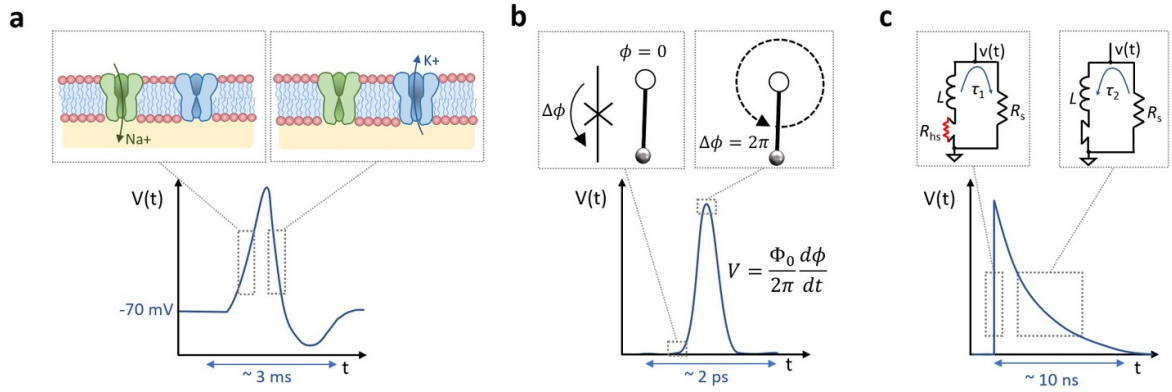

Figure 3. Nonlinearities used to generate spiking in biological neurons and superconducting neuromorphic circuits. (a) Simplified schematic of a biological action potential. Insets highlight the role of the sodium and potassium ion channels, which dictate the rise and fall of the membrane potential, respectively. (b) Schematic of a quantized voltage pulse in a JJ. Insets show how the winding of the phase difference by 2π produces a discrete pulse according to the Josephson voltage relation. (c) Schematic of a relaxation oscillation in a superconducting nanowire shunted by a resistor. A spike is produced as the nanowire switches from the superconducting state to the normal state, diverting the bias current.

a node into a frequency, followed by a set of frequency-to-current converters to change this into a current.

Other ways of representing the sum in figure 2(a) are possible in superconducting electronics. Chiarello *et al* [12] use fluxes to represent the state, mutual inductance to represent the weights, and superconducting quantum interference device (SQUID) loops to represent neurons. The product and summing functions are then intrinsic to the laws of induction. Once again, the critical current of a JJ forms a sharp threshold. Chiarello *et al* [12] were able to experimentally realize a two-layer network to perform an exclusive-OR (XOR) function.

While the switching of a junction provides a sharp nonlinearity, it does result in a power dissipation of roughly $\Phi_0 I_c/2\pi$ for each SFQ pulse. In a scheme that realizes low power dissipation similar to that in adiabatic quantum-flux-parametron logic, Schegolev *et al* [13] create a non-switching processing node that nonetheless still attains a sigmoid thresholding function. Weighting occurs with transformers, similar to Chiarello *et al* [12] Utilizing the transfer function of the RF-SQUID, these authors design a flux-to current converter with asymmetric inductance that nicely rounds into a sigmoid function. The result is that they are able to represent both weights and output states with current and obtain ultra-low power dissipation.

Other neuromorphic architectures take a more biorealistic approach by communicating with electrical pulses similar to action potentials. In hardware, pulses are usually formed through the intrinsic nonlinearity of a device or simple circuit, similar to the nonlinearity responsible for creating action potentials in biological neurons. Figure 3(a) shows a simplified schematic of a biological action potential, highlighting the two voltage-gated ion channels that dominate its behavior [3]. The rising edge of the pulse is created by an influx of sodium ions that raises the membrane potential after the sodium channel opens in response to a sufficiently large input signal. The enhanced potential eventually causes the potassium ion channel to open while the sodium channel closes, leading potassium ions to flood out of the cell and effectively reset it. While other ions also play a role in the creation of an action potential, the nonlinear opening and closing of the sodium and potassium channels are broadly responsible for its overarching behavior.

Several superconducting spiking platforms have been inspired by the similarities between action potentials and the nonlinearity intrinsic to JJs. The voltage across a JJ depends on its phase difference, which ranges from 0 to 2π due to the sinusoidal nature of the current-phase relationship. Figure 3(b) illustrates how the phase winding can be visualized as a pendulum that creates a quantized voltage pulse as it completes a full 2π revolution. This natural spiking mechanism along with operation at tens to hundreds of gigahertz and low power dissipation, make JJs an appealing candidate for spike-based computing architectures.

Figure 4(a) shows a simplified circuit schematic of a JJ-based neuron first proposed by Crotty *et al* [3]. In this design, two JJs are connected in a superconducting loop and act analogously to the sodium and potassium ion channels of a biological neuron. In response to an input signal, the 'sodium' JJ fires and adds an influx of counterclockwise current into the loop, which triggers the 'potassium' JJ to fire in the opposite direction and remove the loop current. Simulations suggested that the JJ neuron is capable of reproducing many characteristics of biological neurons, such as a firing threshold, a refractory or 'resting' period between spikes, and a spiking frequency response that depends on the magnitude of junction damping. Additionally, the JJ neuron uses an estimated 10^{-17} J per spike, in comparison to 10^{-8} J for silicon-based platforms, highlighting the competitive energy performance of superconducting architectures. More recently, the JJ neuron was extended to include Josephson transmission line axons and capacitive synapses whose strengths are controlled by flux-tunable SQUIDs [5, 14]. Experiments demonstrated how mutual coupling between two of these JJ neurons can be used as a foundation for studying complex synchronization dynamics in biological systems (see section 4.3), emphasizing the

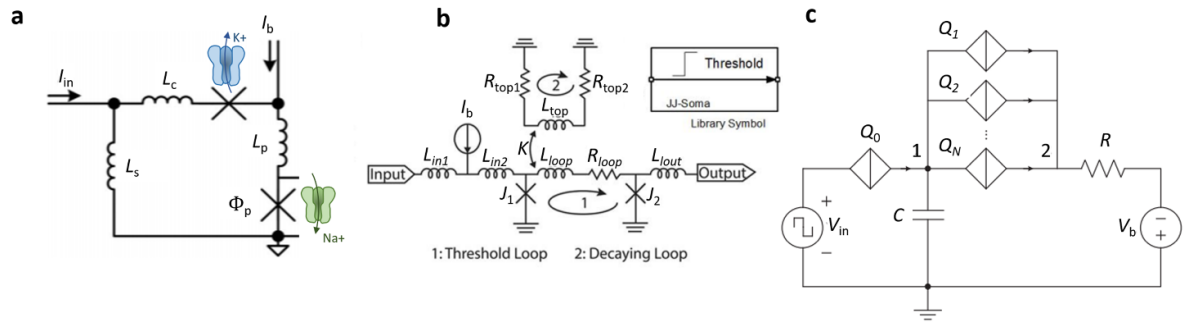

Figure 4. Junction-based spiking neurons. (a) Circuit schematic of the JJ neuron adapted from [3], which uses two JJs analogous to the two ion channels in a biological neuron. (b) Diagram of a spiking neuron based on JJs with a thresholding loop. Figure adapted from [15]. (c) Circuit schematic of an integrate-and-fire neuron using quantum phase-slip junctions. Figure adapted from [15].

dual use of these devices as both an energy-efficient computational platform and as a tool for exploring theories of how biological neurons interact.

Other JJ-based models have also been proposed. Figure 4(b) illustrates a spiking neuron with a two-pulse firing threshold. In this design, two JJs form a 'threshold loop' that temporarily stores an input pulse for a duration set by a resistive 'decaying loop' inductively coupled to it, thereby setting the maximum amount of time between two successive pulses so that their sum exceeds the firing threshold [15]. Measurements of fabricated somas using DC-to-SFQ and SFQ-to-DC circuits to perform read and write operations validated this firing mechanism for different thresholds. Estimates suggest the energy per pulse is on the order of 10^{-19} J.

Another recently proposed design relies on quantum phase-slip junctions, one-dimensional nanowires that are considered the exact dual to JJs since they produce quantized current or charge pulses in units of $2e$, where e is the electron charge [16]. As shown in figure 4(c), the firing of an input quantum phase-slip junctions accumulates charge on a capacitor, building a potential that eventually exceeds the threshold of the n output quantum phase-slip junctions connected in parallel. This design is an example of an integrate-and-fire neuron, since a specific number of input pulses must be accumulated on the capacitor before the parallel quantum phase-slip junctions fire together and generate an output spike. Although the quantum phase-slip junctions neuron has yet to be experimentally realized, its energy per spike is estimated to be on the order of 10^{-21} J, suggesting its potential as an energetically competitive technology.

In addition to JJs, superconducting nanowires have also been used as a hardware platform for artificial spiking neurons. Unlike the current-phase relationship in JJs, spiking is created from relaxation oscillations that occur when a nanowire is placed in parallel with a resistive shunt. As shown in figure 3(c), as the nanowire switches from the superconducting state to the thermally resistive state, the bias current oscillates between the shunt and nanowire with a time constant dictated by the nanowire's high kinetic inductance. Although slower than JJs, these relaxation oscillations are potentially more robust to noise and better suited for fan-out due to their ability to support high impedances, making them appealing candidates for the large parallelism required by neural networks or for interconnects between superconducting neuromorphic circuits and classical hardware. A recently demonstrated nanowire neuron follows the same design as the JJ neuron in figure 4(a), with the junctions replaced by nanowire relaxation oscillators [17]. Measurements of a fabricated neuron showed that it reproduced multiple bio-realistic behaviors, including a refractory period and firing threshold, while simulations of nanowire neural networks suggested that they may be used as inference circuits for pattern recognition [18]. As discussed in more detail in section 4.4, simulations also explored how the stochastically varying switching current in nanowires could be used for testing theories about the role of probability in biological neuron dynamics.

2.2. Synapses

In a biological context the synapse modulates the strength of the signal between two neurons. To describe the synapse, it is then necessary to also discuss the two neurons that it connects. These neurons can be defined as the presynaptic neuron (where the input signal to the synapse originates) and the postsynaptic neuron. On the presynaptic side of the synapse, an action potential causes the release of a neurotransmitter that binds to receptors on the postsynaptic side of the synapse. This release of neurotransmitter and the subsequent binding to receptors opens ion channels across the synapse. Depending on the nature of the synapse, this ion flow can either have an excitatory or inhibitory effect on the post-synaptic neuron. The strength of this connection between neurons can change over time and often exhibits Hebbian learning, where neurons that fire together tend to strengthen their connections [7, 19]. In the context of artificial neural networks, the variable weights applied to the node inputs are inspired by these synaptic connections. They are used to determine the strength of the incoming connection and can typically be either positive or negative in value. These synaptic weights in a neural network are then trained to perform certain tasks with an algorithm such as back-propagation [20]. In neuromorphic systems, the synaptic weighting is typically either carried out

by an analog (or quasi-analog) hardware device, or by accessing stored memory values in digital implementations. In both cases the devices or memory values are typically trainable, which can adapt the hardware to different problems. If the synaptic weights are not trainable then the hardware can typically perform only inference (such as image classification) for one particular task. While such hardware is not a general processor, it can be thought of as an application-specific integrated circuit (ASIC), which can be extremely fast and energy efficient for a given task.

2.2.1. SQUID synapses. A synapse can be formed by a two-junction SQUID at the end of a Josephson transmission line (JTL). In a JTL, propagation occurs when sequential junctions are triggered to undergo a 2π phase rotation as the pulse is propagated to the next stage. The peak voltage of each junction is given approximately by I_cR_s, where I_c is the critical current of the junction and R_s is the shunt resistance. Replacing a junction with a SQUID gives it effectively a variable critical current, depending on the flux in the SQUID loop. If the SQUID replaces the final junction in the JTL, then the size of the pulse leaving the JTL will be dependent on the SQUID flux. If the JTL represents an axon, then the end of the axon will act a synapse, with the pulse leaving the SQUID analogous to the post-synaptic current. Since the size of the post-synaptic current will depend on the flux in the loop, then this flux will represent the weight of the synapse. This scheme was utilized by Segall *et al* in a scheme with mutually coupled neurons. The flux through the SQUID loop was able to change the coupling to cause a phase flip bifurcation as seen in figure 10. In addition, biased SQUIDs have also been proposed for the synaptic element in adiabatic flux based superconducting systems [21].

2.2.2. MJJ synapses. Superconducting spintronic devices can use the interaction of ferromagnetic and superconducting order parameters to create new functionality [22–24]. Recently there has been significant effort in this area and many interesting devices are currently being investigated, such as π JJs [25], spin triplet JJs [26], and pseudo spin-valve JJs [27]. As an understanding of how to make these devices grows, new novel devices can be made in which a ferromagnetic layer in a JJ can be used to modulate the phase and/or amplitude of the Josephson critical current I_c. However, one potential issue with these devices is that the energy and time scales required to manipulate the magnetic layer are typically much larger and longer than those of SFQ-based JJ logic. From basic size scaling arguments, it can been seen that the interaction of a magnetic moment with an SFQ pulse requires a small cluster of magnetic material with a diameter on the order of a few nanometers.

Recently, a magnetic Josephson junction (MJJ) with nanoscale magnetic clusters in the barrier was demonstrated in which the magnetic order affects the critical current of the junction [6]. These non-exchange-coupled magnetic nanocluster are small enough that above the blocking temperature of 50 K the thermal energy of the system is enough to change the direction that the net magnetic moment of any given nanocluster points. When all of the magnetic moments point in the same direction, this defines the fully ordered magnetic state for the junction. If the magnetic moments are randomly oriented, the junction is in the magnetically disordered state. In these recently demonstrated devices the magnetic nanoclusters are made from Mn and have a size of about 4 nm in diameter, and are quite dense with approximately 20 000 clusters per square micrometer of junction area. This means that there should be a large number of states that exist where the number of magnetic nanocluster pointing in the same direction is somewhere between fully ordered and fully disordered. Measurements of these devices have shown that the intermediate states can also be accessed in these devices, making them appealing candidates for synapse-type devices in superconducting circuits.

Mn nanocluster MJJs have demonstrated a change in I_c from 1 mA to 10 μA by changing the magnetic order. In order to modify the magnetic orientation of the state one can use short (100 ps) electrical pulses while the chip remains nominally at 4 K. The use of short electrical pulses in the absence of any applied magnetic field tends to disorder the magnetic clusters, whereas the application of short electrical pulses in the presence of a small magnetic field tends to order the magnetic clusters. The applied ordering field can be made small enough that only devices that see the electrical pulse have their magnetic order affected. In addition, because the blocking temperature is roughly 50 K, defluxing the circuit does not affect the magnetic order in the MJJs.

Figure 5 (originally from [6]) part (a) shows the characteristic voltage versus current curve for the magnetically disordered state of a 10 μm diameter circular clustered MJJ. The data (blue circles) are fit with a resistively shunted junction model [28]. The fit gives a value of $I_c = 1.30$ mA $\pm$ 0.02 mA and $R_n = 1.32$ mΩ $\pm$ 0.02 mΩ. Figure 5(b) shows the characteristic voltage versus current curve for the same device now in the magnetically ordered state. The Resistively shunted junction fit gives a value of $I_c = 0.08$ mA $\pm$ 0.02 mA and $R_n = 1.30$ mΩ $\pm$ 0.02 mΩ. The data are plotted with the same current range for clarity of comparison. It is also worth noting that change of magnetic order does not affect the normal state resistance of the junctions, as is expected. Figure 5(c) shows data from a 10 μm diameter clustered JJ. The device started in a magnetically disordered state and was successively ordered by short (240 ps, 11 pJ) electrical pulses applied in the presence of a 20 mT external magnetic field. Figure 5(d) shows (red squares) the value of the electrical pulse energy required to change a device of a given size from magnetically disordered to magnetically ordered in the presence of a 20 mT magnetic field. The calculated dissipation energy of an SFQ pulse from these junctions ($\approx I_c\Phi_0$) is shown with blue circles for the magnetically disordered state.

The clustered MJJs with variable critical current can be used in different ways for biologically inspired superconducting circuit design. In one potential architecture, one can lower the critical current value to lower the threshold of the MJJ for emitting SFQ pulses to tune the spiking behavior. It is worth noting that, in this context, the devices may offer a way to adjust threshold values in a neuron (or soma) type circuit in

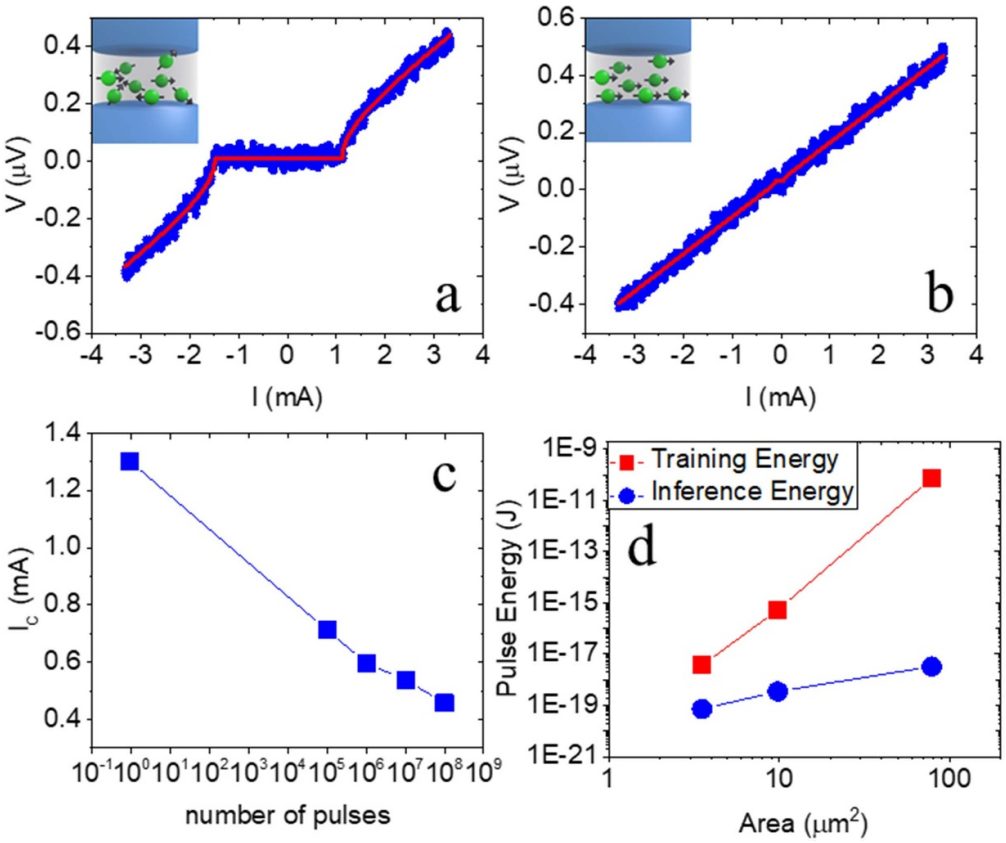

Figure 5. Data (blue circles) of the voltage versus current characteristics taken at 4 K on a 10 μm diameter nanoclustered MJJ along with the fit to the resistively shunted junction model (red line) shown for the magnetically disordered state (a) and the magnetically ordered state (b). (c) MJJ critical current taken at zero applied magnetic field vs. the number of ordering pulses applied in a 20 mT applied field, line is a guide to the eye. Figure (d) electric pulse energy required to fully order the MJJ in a 20 mT applied field (red squares), and calculated SFQ pulse energy of the MJJ (blue circles) vs. cross-sectional area.

addition to being used in synapse type circuits. In another potential architecture, the device can be used as a variable inductor, since the Josephson inductance scales as $1/I_c$. For example, on can use a clustered MJJ on one leg of an inductive divider and use the other leg to couple current (or flux) into a postsynaptic neuron circuit. It has also been suggested that MJJs can be used with adiabatic superconducting circuits to perform the synaptic function with potentially even lower energy costs [29].

While these devices are very promising, they are currently still very new and as such need to be tested on a larger scale. One of the primary drawbacks of these devices is the current requirement of a 400 °C annealing step. While such temperatures are compatible with CMOS processing, JJ processing typically does not have adequate thermal budget. As such, this step limits the use of these devices to layers below any of the JJ layers in the stack. Because the electrodes are Nb, and the barrier is quite thin (<5 nm) and the devices themselves would have the thermal budget for additional layers on top, there is no fundamental reason that this will not work. However, incorporating a new layer with new materials at or near the bottom of the fabrication process will likely take some significant development. Demonstrations with a larger number of clustered MJJs working together in a circuit will likely be needed to get a better idea of device-to-device variations before such development is likely to be undertaken. One other issue with the previously demonstrated version of these devices is that the I_cR_n product is quite low. While overdamping in and of itself is not so problematic, it would potentially limit the speed of biologically inspired circuits that use these elements. Further work on the materials side of these devices could mitigate these potential difficulties, with a higher I_cR_n product and reducing or eliminating the annealing step being particularly helpful. Another future direction for these devices would be the incorporation of a spin-polarizing layer within the device. If successful, such an addition could eliminate the need for any applied magnetic field and could enable the increase or decrease in I_c depending on the direction of the current flow.

2.2.3. Optoelectronic synapses. The strengths of superconducting electronics for neural information processing extend to very large cognitive systems. Communication is crucial in neural computing, and to interconnect systems with millions or even tens of billions of neurons at the scale of the human brain, photonic communication may be advantageous. Superconducting single-photon detectors bring an extraordinary advantage in this regard, as one quantum of electromagnetic radiation is then sufficient for a neuron to

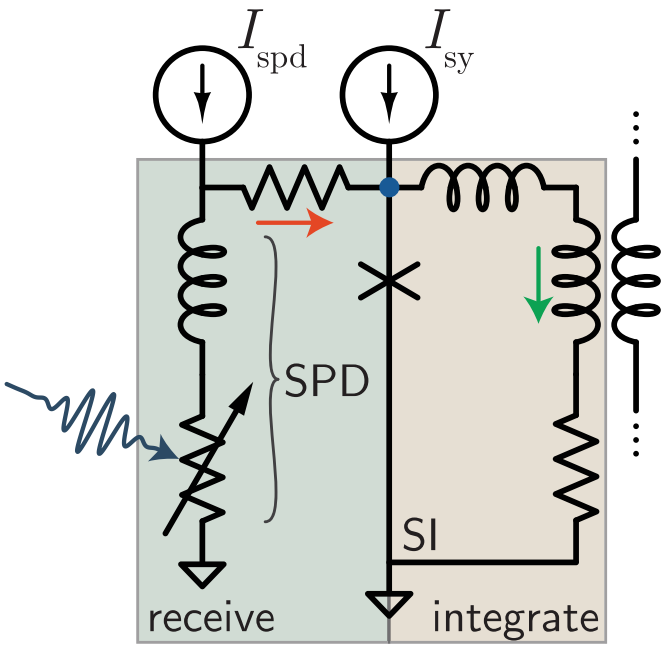

Figure 6. Superconducting optoelectronic synapse converts photonic communication events to integrated supercurrent. The single-photon detector is labeled SPD and biased by a direct current, I_{spd}. The synaptic bias current, I_{sy}, controls the amount of current added to the synaptic integration loop (SI) with each synapse event. Many such synapses can be coupled to a common neuron through transformers, as described in section 3.1.

communicate to a synapse. By contrast, if semiconductor photodiodes are used in conjunctions with MOSFETs as syanpses to receive photonic communication events, roughly one thousand photons are required for communication [30]. Superconducting detectors therefore dramatically reduce the optical power required for communication between distant neurons in a large network, reducing the required brightness of each light source by a factor of one thousand.

Synapses based on single-photon detectors operating in conjunction with JJ circuits have been designed [31–33], and demonstrations of the basic principle have been accomplished [34] using thin-film constrictions in place of tunneling-barrier JJs [35]. One manifestation of such a synapse comprises a superconducting-nanowire single-photon detector in parallel with a JJ embedded in a flux-storage loop (figure 6). In this manifestation, the synaptic-firing junction (J_{sf}) is current-biased below its critical current. When the single-photon detector (SPD) detects a photon, all the current from that branch of the circuit is diverted to J_{sf}, driving that junction above I_c, causing the junction to produce a series of fluxons. Those fluxons are added to the synaptic integration loop (SI), which accumulates a circulating current with magnitude dependent on the recent history of synaptic activity. This signal decays with a time constant set by the passive circuit parameters: $\tau_{si} = L_{si}/r_{si}$, much like the charge leaking off a biological neuron's membrane capacitance. The signals from many such synapses can be coupled to a common neuron through transformers.

While such optoelectronic synapses have been designed to enable communication across large spiking neural networks with optical signals at the single-photon level, they also offer the possibility of event-based adaptation and unsupervised learning mechanisms. The synaptic weight of such a synapse is set by the current bias to the synaptic firing junction, I_{sy}. When this current is low, the sum of the bias to the SPD and the JJ ($I_{spd} + I_{sy}$) will only slightly exceed I_c, so the junction will be driven above I_c briefly, and will therefore produce only a small number of fluxons. In this case, a small amount of current is added to the SI loop, and the synaptic weight is weak. When I_{sy} is very close to I_c, the sum $I_{spd} + I_{sy}$ will be well above I_c and will stay above I_c for nearly the entire recovery time of the SPD. In this case, many fluxons will be added to the SI loop, and the synaptic weight is strong. Such circuits can be designed such that a small synaptic weight produces close to a single fluxon, whereas a strong synaptic weight produces over one thousand. In such a circuit, the synapse may have a bit depth of 10. Plasticity and learning can be accomplished by coupling other JJ-based circuits to the bias I_{sy}. Designs for circuits capable of implementing STDP were presented in [31, 32], and further extensions to achieve short-term plasticity and metaplasticity are possible based on the same principles.

While the recovery time of the SPD (10 ns) is slower than the switching time of a JJ (10 ps), it is the high speed of JJs that allows analog operation in this context. With this form of optoelectronic neuron, the maximum neuronal firing rate must be reduced due to the SPD recovery time, while fast JJ processing is leveraged for computations performed within each neuron. One proposed architecture uses dendritic trees with dendrites very similar to typical JJ neurons [33, 36]. Sophisticated neurons thus envisioned are closely akin to the elaborate neural processors comprising human cortical neurons [37, 38], wherein dendritic computations appreciably augment neuronal functionality [39].

2.2.4. Other synapses. There have been several other suggested synaptic elements for different superconducting neuromorphic systems. These include building a system based on adiabatic superconducting cells that can perform the sigmoid transfer function [13] and cells that approximate the derivative of the sigmoid activation function [40], in order to perform a learning rule such as resilient back-propagation [41]. In addition, circuits have been demonstrated that have used SQUIDs for neurons and the conductance of resistors between the SQUIDs to perform the synaptic weighting function [42, 43]. There have also been simulations of a superconducting neuromorphic system that use quantum phase-slip junctions combined with magnetic JJs for the synaptic operation [44]. In addition to these analog-type synaptic elements, a binary synaptic element combined with a threshold neuron has also been demonstrated [15].

3. Connectivity

In addition to choices regarding soma and synapse implementations, neuromorphic architectures are also defined by

the interconnection scheme used to enable communication between the neurons in the network. In biological neural networks this interconnection functionality is provided by axons, synapses, and dendrites. The output signal from a spiking neuron is carried along a branching axon that forms synapses with downstream neurons. The axon is the biological component that accomplishes fan-out and provides the connections between neurons. Synapses thus formed may connect to the receiving neuron's dendritic tree, the soma itself, or the axon initial segment. Fan-in is performed primarily in the dendritic tree [45]. In biological systems, the response of the dendrites is active and nonlinear, resulting in stages of information processing between the synapses and the neuron cell body [39]. This dendritic processing enables neurons to detect and respond preferentially to certain patterns of activity, such as coincidences and sequences [46], as well as to perform intermediate thresholding operations on groups of inputs [47] and influence learning [48, 49]. Such functions appear possible with superconducting circuits [33]. However, these functions are typically omitted from hardware implementations (with exceptions [50]) as the complexity of networks without this functionality is sufficient to generate rich dynamical activity. In superconducting neuromorphic architectures this interconnection functionality is provided by the wires that connect the neurons and the circuits that handle fan-in and fan-out. However, there are many ways to implement and organize these components. In general, a superconducting neuromorphic architecture's interconnection scheme is shaped by three main choices. First, how neurons should be physically connected in the system. Second, how spikes should be represented in the system. And third, how the spikes should be interpreted by the system.

At the highest level this interconnection architecture can be either fully dedicated or time multiplexed. Fully dedicated architectures provide a physical wire for each connection that is needed between neurons. This approach is the most straightforward in terms of design and can natively preserve analog timing relationships within the network. However, because physical wires are needed for every connection that might exist between any two neurons in the system, the wiring-density requirements of this approach can become excessive when the number of neurons in the system becomes very large. Connections can be virtually eliminated by setting synapse weights to zero, but the physical wires will remain. This challenge can be somewhat addressed by implementing a more targeted network architecture that does not include all of the possible connections between neurons. In other words, a specific network configuration or a class of networks could be implemented rather than hardware capable of supporting any network configuration. In addition, because the number of physical connections between neurons can be high even in targeted implementations, support for high fan-in and fan-out is needed.

Alternatively, time-multiplexed architectures share physical wires between neurons in order to reduce the wiring density required when implementing systems with very large numbers of neurons. This approach often makes use of a digital network with routers to provide the shared connectivity. Address event representation (AER) packets are used to specify where spikes should be routed. The AER packet can also include timing information so that timing relationships between neuron firing events can be preserved. However, recreating these timing relationships from AER packets is significantly more complex than the native support found in fully dedicated architectures. In addition, large amounts of memory may be needed to store the addresses of post-synaptic neurons and to store the routes for the routers in very large systems. Novel memory technologies and memory system architectures will likely be needed to address this need. Dense superconducting digital logic will also be needed to provide the functionality required by a digital network. Suitable technologies have been recently demonstrated as a result of research programs such as the IARPA C3 program, but digital networks have not yet been implemented using these technologies.

In superconducting neuromorphic systems, the method of communication can also vary and has a significant impact on the capabilities of the system. For instance, SFQ pulses can be used to represent the spikes that travel between neurons. This allows for relatively straightforward communication between the components of the system because every circuit is based around the same signal type. This enables an uncomplicated interface between analog and digital components in the system. Alternatively, multiple SFQs can be used to represent a single spike event. In implementations involving only JJ-based circuits, generating multiple SFQs for every spike can require prohibitively complex circuitry. However, in optical or nanowire implementations, multiple SFQs can result natively from signals traveling between neurons. An advantage of multiple SFQ approaches is that they can support greater degrees of fan-in and fan-out thereby enabling fully dedicated systems with more neurons.

Finally, the spikes themselves can be utilized in a rate-coded or temporal-coded fashion. In a rate-coded system, only the number of spikes that are received per unit time is considered by the activation function. In other words, the neuron will fire if enough spikes are received before some deadline such that the sum of the weight of those spikes exceeds some threshold. In superconducting electronics, such a rate coded system can be built in an analog fashion by utilizing a loop to aggregate flux quanta from all of the incoming spikes that are received over some period of time. The same function can also be realized in a digital fashion by using superconducting digital logic to accumulate the weight of all incoming spikes. Another way to interpret the spikes is to consider not only the number of spikes received but also the timing relationships that exist between them. This is referred to as temporal coding. The timing relationships between spikes can be natively preserved in a fully dedicated, free-running network implementation. In this architecture, the neurons will receive spikes at different times depending on when the pre-synaptic neurons fired. Refractory periods and leaky soma circuits can then ensure that different patterns of incoming spikes result in output spikes that occur at different times. Again, similar functionality can be implemented using digital circuits that use the timing information in AER packets to establish when incoming spikes should be applied to the soma.

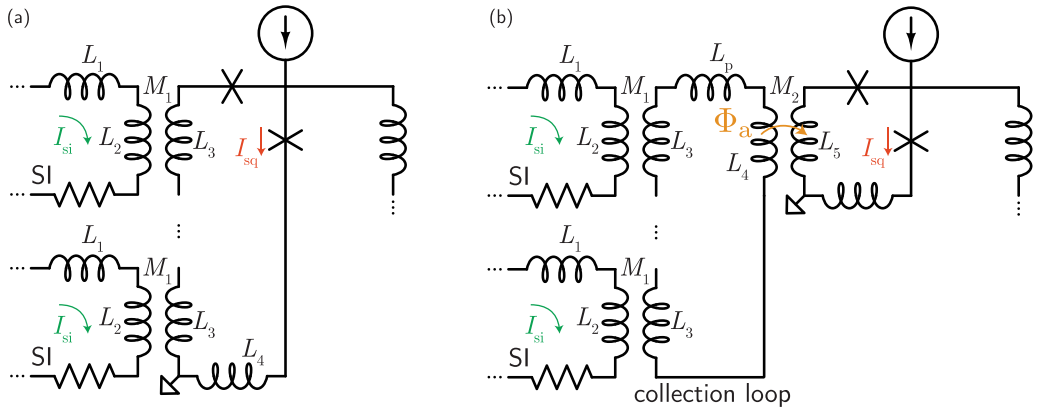

Figure 7. Transformer circuits for fan-in. (a) Direct fan-in circuit with multiple input synapses coupled to a SQUID performing the role of the soma. (b) Synapses coupled to a common collection coil.

3.1. Fan-in for superconducting electronics

Fan-in in semiconductor-based neural circuits is straightforward, owing to the large impedance of MOSFETs. The source-drain current from many MOSFETs can be directly summed via Kirchhoff's current conservation law without significant cross talk or leakage current. By contrast, the low impedance of superconducting wires and JJs results in appreciable current leakage through unintended paths to ground when multiple wires are directly connected to a common node. For this reason, fan-in leveraging mutual inductance has long been preferred in superconducting neural circuits [51, 52].

Examples of circuits accomplishing fan-in through transformers are shown in figure 7. We consider two primary cases, and in both a DC SQUID is employed as the soma, which sums input flux and produces one or more fluxons when the current through one of the JJs exceeds I_c. In figure 7(a), the outputs from synapses are directly coupled to the SQUID loop, whereas in figure 7(b) a collection loop is used to passively sum the input flux. The direct configuration reduces parasitic inductance, making the SQUID more sensitive to small inputs, but this implementation leads to a practical fan-in limitation owing to the fact that the total inductance of the SQUID loop must be kept relatively low to ensure $\beta_L = 2LI_c/\Phi_0$ stays close to one, as is desirable for low-noise SQUID operation [53]. The direct method of figure 7(a) is preferable in the case where the action potentials are single SFQ pluses, whereas the collection loop method shown in figure 7(b) may be preferable in the case where the action potentials contain many flux quanta, so that the SI loops have large inductance and can couple strongly to the SQUID. The single SFQ approach is expected to be better in terms of energy efficiency and speed, whereas the multi-SFQ approach allows analog post-synaptic signals which may bring computational benefits while still maintaining suitable performance for energy and speed. In this section we look closely at how both methods scale to large fan-in, indicated by the number of synapses N.

We first consider the single SFQ case and the circuit of figure 7(a). Using the equations of mutual inductance, the flux coupled into the SQUID loop Φ_{sq} from a single synapse is simply equal to MI_{si}, where I_{si} is the synaptic input current and M is the mutual inductance $M = k\sqrt{L_2 L_3}$, and k is the inductor coupling constant. This flux results in a SQUID current I_{sq} given by the following:

$$I_{sq} = \frac{k}{N}\sqrt{\frac{L_2}{L_3}}I_{si}. \tag{1}$$

Here we have assumed that the self-inductance of the loop is NL_3, ignoring the L_4 term and the Josephson inductances of the junctions in the limit of large N. In order to trigger an action potential, this current needs to be larger than the critical current of the junctions in the loop (I_{c3}), or conversely the size of the flux Φ_{sq} must be larger than $\Phi_0(\beta_L/2\pi)$.

If SFQ pulses are being used for action potentials, two adjustments to this expression for I_{sq} can be made. First, since the current I_{si} is from an SFQ pulse in the synaptic input loop we can write:

$$I_{si} = \frac{\Phi_0}{L_1 + L_2}. \tag{2}$$

In addition, the critical current of the junction in the synapse input loop (I_{c2}) can be made larger than the critical current in the SQUID loop I_{c3}, increasing the SQUID current by a factor of (I_{c2}/I_{c3}). (Note: If one does in fact do that it is important that the SQUID neuron loop be followed by an amplifying JTL to bring the current levels back to the level of I_{c2}.)

To explore the limit of large N, some other design assumptions must be made. First, how many synapses P (where $P < N$) should be active at a given time to fire the post-synaptic neuron? One can consider the case as low as $P = 1$ (a single synapse triggers an action potential) up to something like $P = 0.2$ N (20% of the synapses firing at one time). Second, how large must the SQUID current I_{sq} be compared to the critical current I_{c3}? One can use an external bias current to compensate for a small SQUID current, but this has limits due to fabrication spreads. A recent work [54] explored the limit of large-N fan-in requiring that $P = 1$ (by far the worst case) and $I_{sq} = 0.2I_{c3}$ (also very conservative). Using fabrication parameters achievable with today's technology, a fan-in of 100 was simulated (see equation (8) and figure 10) [54]. Considering

that the majority of neurons in the brain are granule cells with fan-in of around four [55], this fan-in situation appears favorable. Further, fan-in at much higher levels can also be achieved with slightly modified circuit concepts, as described below.

When using SFQ pulses to communicate from neurons to synapses and from synapses to neurons, it is advantageous to reduce the I_c of the JJs in the SQUID comprising the neuron cell body so that a relatively small fraction of active synapses can drive the neuron above threshold, as described above. It is also illuminating to consider the other end of the spectrum, wherein synaptic integration loops (SI in figure 7) may have large inductance and be used to couple strongly to a SQUID soma. The response of a SQUID is periodic in applied flux, and in general the integrated current from many synapses may couple an amount of flux into the SQUID that drives the circuit through multiple periods of its response function. Next we consider the scenario in which we design all transformers so that the maximum applied flux to the soma is limited to $\Phi_0/2$ so the response of the neuron is a monotonically increasing function of applied flux. In the above calculation we constrained the inductance of the synapses to ensure SFQ operation, and this inclined us to consider synapses directly coupled to the SQUID loop to reduce parasitic inductance as well as to use JJs with reduced I_c in the SQUID relative to the synapses. Here we consider allowing the synaptic inductors to take any value, but we consider the fan-in ramifications of limiting the total applied flux to the soma to $\Phi_0/2$.

When the synaptic inductances (L_1 in figure 7) are allowed to take larger values than required for SFQ operation, it is not necessary to couple synapses directly into the soma. In this context, one can use an intermediate collection coil, as shown in figure 7(b) to passively sum all synaptic signals and couple them into the soma SQUID with one transformer. In general, the net flux to the soma (Φ_a in figure 7(b)) can far exceed Φ_0, causing the neuron to have a periodic response with applied flux. Each synapse will have a maximum current at which the loop saturates, I_{sat}. The maximum flux to the SQUID (Φ_{max}) occurs when all synapses are saturated. We can enforce the condition $\Phi_{max} = \Phi_0/2$ through a particular choice of inductors comprising the transformers, introducing a constraint on L_2 and L_4 in terms of the junction I_c and the number of synaptic inputs to the collection loop. For $I_c = 100$ μA, this constraint can be satisfied with L_2 and L_4 between 10 pH and 1 nH across a wide range of numbers of inputs. These inductances are straightforwardly achieved for synapses with large self-inductance (L_1), but this inductance constraint cannot be satisfied in general while also satisfying the constraint that $L_1 + L_2 = \Phi_0/I_c$, as required for SFQ operation. Thus, if SFQ operation is desired, it is difficult to also enforce the constraint that the maximum applied flux is limited to $\Phi_0/2$.

Fixing $\Phi_{max} = \Phi_0/2$ through the choice of L_2 and L_4 has important consequences for fan-in. In this context, the fraction of synapses that must be active to drive the neuron to threshold depends primarily on the bias point of the soma SQUID. Additionally, for typical values of this bias point ($0.6I_c \leqslant I_b \leqslant 0.9I_c$), a relatively large fraction of synapses must be active to reach threshold. This may be undesirable, as it will require extensive network activity before any neuron will fire, rendering sparse coding intractable. One method to address this challenge is to use an active dendritic tree. In this case, the circuit block of figure 7(b) is repeated in a tree graph. Now each collection loop and SQUID define a dendrite. Consider the homogeneous case wherein each dendrite receives n inputs, and there are H tiers to the tree. The first tier at $h = 0$ is the final receiving SQUID of the soma itself, intermediate tiers involve dendrites coupled into other dendrites, and at $h = H$ are the synapses coupled into dendrites. The total number of synapses is given by $N = n^H$. We assume all dendrites in the tree have the same values of I_c and that the transformers are designed to satisfy the condition $\Phi_{max} = \Phi_0/2$ so that all responses are monotonically increasing functions of the input flux. Given these design choices, it is found through numerical integration of the SQUID equations [53] that an active dendritic tree can significantly decrease the fraction of synapses that must be active to drive a neuron to threshold. We define the number of saturated synapses required for threshold as P, the total number of synapses as N, and the depth of the dendritic tree as H. The case of a point neuron corresponds to $H = 1$. For example, a point neuron with transformers designed to limit flux $\Phi_a \leqslant \Phi_0/2$ biased at $I_b = 0.7I_c$ with $N = 1000$ would require 71% of synapses to be saturated to drive the neuron to threshold, while using a tree with $n = 6$ and $H = 4$ would require 25% synaptic activity. For a very large neuron with $N = 10,000$, $n = 6$, and $H = 5$ would require 18% of synapses to be active, and if $I_b = 0.9I_c$, with $H = 4$ and $n = 20$ drops P/N down to 1.3%.

This brief analysis of fan-in is intended to illustrate several aspects of superconducting neural circuit fan-in. One result is that using synapses with SFQ inputs to the neuron is incommensurate with using a collection loop and requires using different junction I_c for neurons and synapses. The proper neuron I_c depends on the number of synaptic inputs, N. If one starts analysis at a different point and requires the maximum applied flux be limited to $\Phi_0/2$ to ensure a monotonic response, it is likely necessary to use a dendritic tree to reduce the number of synapses that must be active for threshold. If one prefers to utilize a single SQUID design with fixed I_c for many synapses, dendrites, and neurons without modifying the design based on the number of inputs, then a collection coil is advantageous. However, this limits the ability to utilize SFQ synapses, and instead inclines the designer to employ larger synaptic inductors, leading to analog signals stored in synaptic integration loops and communicated to the neuron cell body. It is possible that complex networks will leverage multiple of these design concepts, as intelligence is observed to benefit from a diversity of neuron types [56]. Additionally, due to the significant implications of limiting applied flux to $\Phi_0/2$ to ensure monotonic response, consequences of allowing a periodic response from dendrites and neurons merits investigation in a network context. If this behavior can be accommodated or may be useful, design of circuits for fan-in can be adjusted accordingly.

3.2. Fan-out for superconducting electronics

When building toward large cognitive systems, the architecture of the brain serves as an indication of the expected hardware requirements. Most of the neurons in the human brain reside in the cerebellar granule-cell layer. These 50 billion neurons only receive input from about four synaptic connections [55]. Such neurons are thought to participate in associative learning to realize a high-dimensional representation of the information provided by their inputs. This degree of connectivity is straightforwardly achieved with superconducting electronic circuits, wherein two stages of 1×2 pulse splitters can route the output from a single neuron to four recipient synapses with negligible time delay [54].

At the other end of the spectrum, pyramidal cells in the neocortex make thousands of connections, many of which extend over appreciable distances [57]. The networks formed by such neurons are characterized by several graph metrics that are thought to be significant enablers of cortical function. The networks form what is known as a small-world architecture, wherein a high degree of local connectivity is combined with long-range connectivity to realize significant clustering, but also very short average path length across the network [58, 59]. The short average path length enables rapid communication between distant modules whose combined activity may be relevant to processing a given stimulus. Achieving both the high degree of connectivity as well as the significant long-range connections are primary challenges faced by all approaches to neuromorphic hardware, yet these connectivity structures appear necessary to achieve a high topological dimension, which is evidently prioritized in the cognitive circuits of the brain [60].

Achieving this degree of connectivity with electronic circuits remains a primary challenge to scaling of neuromorphic hardware. Making direct connections between semiconductor-based neurons is prohibited by wiring parasitics, and this limitation is partially mitigated through address-event representation (AER) [61–63] digital networks, wherein spike events are given a digital representation as a packet including the time of a spike and a neuron address. A shared communication infrastructure is then used to route spike packets to their intended synaptic connections. This approach has been a key enabling factor in the development of semiconductor neuromorphic systems and allows a great degree of reconfigurability in the network structure, but it comes with a cost. As the number of neurons and their degree of connectivity increase, the size of memory required to store the addresses grows, as does the communication latency.

Similar themes arise when designing interconnection networks for superconducting electronic neurons. While it is the capacitance and resistance of metal wires that makes direct connections between semiconducting neurons challenging, it is the inductance of superconducting wires that requires attention in the superconducting domain. The current associated with a fluxon produced by a JJ varies inversely with the inductance of the wire the JJ must drive ($I_{\mathrm{fq}} = \Phi_0/L$), so repeater junctions must be used to propagate pulses over long distances, forming a Josephson transmission line (JTL). A typical superconducting wire close to a ground plane will have 500 fH μm^{-1} of inductance, and the inductance one would like a JJ to drive is typically less than 100 pH. If one uses wires of 1 μm width, one would expect to place JJs roughly every 200 μm along a JTL. To produce an axon connecting a neuron on one side of a 1 cm $\times$ 1 cm die to a synapse on the other side may require as many as 50 extra JJs. When interconnecting large numbers of neurons in a small-world architecture, increasing the width of the wires to reduce inductance does not bring a net reduction in the number of JJs required in the JTL, because while junctions can be spaced further apart, the destinations also grow further apart by the same scaling factor. As is the case in semiconductor-based architectures, an AER digital network may be a path forward for superconducting neuromorphic systems. This approach is discussed in greater detail in section 4.5.

3.2.1. Large-scale connectivity. While challenging, achieving interconnection of neurons across a die may be possible, and is decidedly worth pursuing [54]. Many exciting applications and research opportunities will be served by high-speed, low-power-density, spiking neural networks of that scale. For those who seek to go beyond that scale, new challenges must be considered. For artificial neurons of any composition, whether semiconducting or superconducting, achieving the complexity of operations supporting learning and adaptation in biological brains requires circuits of much larger spatial extent than the devices grown by biology. Simple area estimates reveal that the 100 billion neurons of the human brain will never fit on a computer chip, nor even a full 300 mm wafer. Yet for large neural networks approaching this number of interconnected neurons, the trend is toward wafer-scale integration [64–66]. At this scale, an axon spanning the wafer will require 10 000 repeater JJs, leading to wiring and current-biasing challenges that appear formidable. It has been acknowledged that such current-biasing challenges must be solved in superconductor digital electronics long before wafer-scale integration is conceived [67]. To make matters more interesting, such wafers are likely to support between one million and ten million neurons, interconnected by billions of synapses—an extraordinary technological goal, but still far short of the cognitive structures inside each human skull.

Optical communication is an enticing approach that could address the challenges of large-scale connectivity in superconducting neuromorphic systems. Optical communication in digital processors is not a new idea [68], and there are sound physical motivations for including optical links in existing computer architectures, even at the chip scale [69]. These motivations include precise clock distribution, energy-efficient long-distance communication, and scalability to multi-chip modules. Given these extensive motivations for optoelectronic integration, the reason modern computer chips do not employ optical communication is simple yet formidable: there is not a known way to integrate light sources with CMOS electronics in a scalable, cost effective manner. If there existed light sources as simple as transistors that could be integrated with CMOS electronics, the hardware and

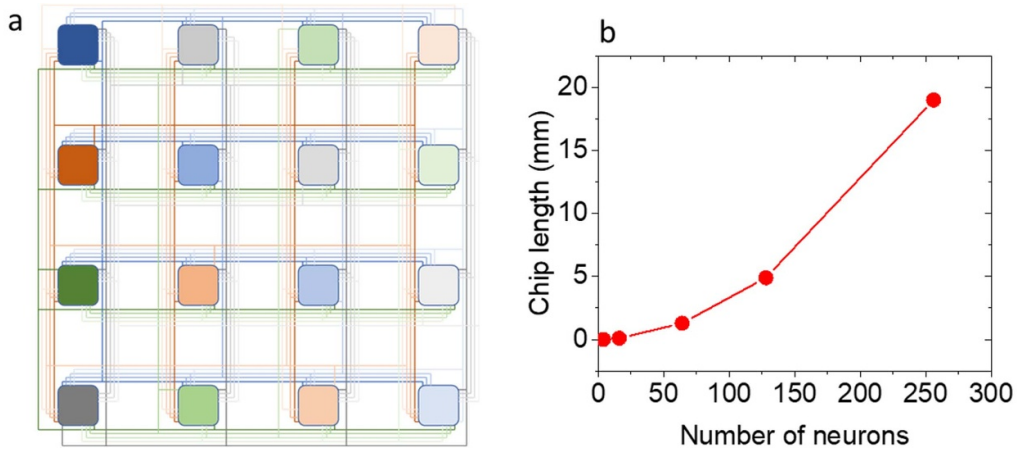

Figure 8. (a) Schematic of a fully connected 16 JJ neuron network (not to scale). (b) Approximate chip length (assuming square) for a fully connected JJ neuron network vs. neuron count.

architectures of modern computing might be appreciably different. Excellent work toward this end [70, 71] notwithstanding, the basic burdens are physical. Silicon has an indirect band gap, so light production is inefficient. Successful light sources are based on compound semiconductors with lattice constants significantly different from that of silicon, so direct growth of light sources on silicon wafers requires processing techniques that, to date, have not been integrated with a CMOS process.

In the superconducting domain, light-source integration remains the primary challenge to optical communication. Yet there are several problems that change the considerations relative to semiconductor optoelectronic integration. First, the use of superconductors enables simple, high-yield detectors that can respond to light levels as low as a single-photon [72, 73], dramatically reducing the output requirements of light sources used by neurons to signal to synapses [74]. Second, superconducting circuits must be operated at cryogenic temperatures, at which point optical transitions in silicon other than slow, band-edge transitions can produce light more efficiently than at room temperature. Third, superconducting electronic circuits can potentially be sputtered and processed on top of III–V semiconductor light sources, enabling greater fabrication flexibility than is possible when integrating with MOSFETs. The prospects for silicon as well as III–V light sources appear more promising when integrated with superconductors than when integrated with semiconductors, potentially enabling multiple routes to light sources as simple as transistors while meeting the needs of large-scale neuromorphic systems [30, 75, 76]. Such systems are envisioned to achieve communication across wafers through passive, dielectric waveguides [77–79], and between wafer-scale modules with low-loss fiber optics [80]. Still, significant hurdles remain to move such light sources from the domain of research-level devices to wafer-scale integrated optoelectronic circuits. In particular, integration of faint, pulsed light sources with superconducting electronic circuits, including JJs and SPDs, at an industrial scale has not been demonstrated. Additionally, while it has been shown that superconducting thin-film amplifiers can produce the voltage necessary for driving a semiconductor light source [81], it remains to be demonstrated that these driver circuits can be accomplished with sufficient speed and energy efficiency to enable scalable systems.

3.2.2. Fully connected networks. Large-scale systems that use only all-to-all connectivity seem very unlikely. However, mixed systems with smaller blocks of fully connected neurons that communicate with other locally fully connected blocks is something that has already been used in CMOS implementations of neuromorphic computing [82–86]. Here we look at how a block of SFQ neurons can be connected in an all-to-all manner and what the basic size requirements are for this type of block.

Figure 8(a) shows 16 fully connected JJ neurons with the associated wiring for an all-to-all connection scheme. Each of the squares here (not to scale) represents the necessary synaptic weight, fan-in, and neuron needed to process the 15 inputs and provide the 1 output. The blocks contain all of the analog components of the neuromorphic SFQ circuit as described in [54]. In this type of architecture, the output and communications are handled as digital SFQ. The wiring between these cells must then typically contain additional junctions to maintain proper impedance. This type of physical layout would most likely require at least 2 wiring layers in order to support the required cross-over points, plus additional wiring layers for biasing the junctions. However, small fully connected analog blocks like these could be an interesting part of a larger system and implement certain problems like best match and traveling salesman type problems [87].

Figure 8(b) shows the nominal size scaling of JJ technology with the number of fully connected neurons. The size scaling is estimated using the following assumptions about the block size and the wiring. The pitch between synaptic elements (contained within the neuron block) is 5 μm and will dominate the size of the block as the number of connected elements is increased since the neuron is 2 JJs regardless of the size of the fan-in. There will also be a JTL amplifier within the block, but again this will be a fixed number of JJs that does

not scale as the number of connected elements is increased. For the estimation, it is assumed that the JJs for the neuron and JTL amplifier will add a constant 10 μm to the sides. The wiring pitch is assumed to be 3 μm. Narrower strip-lines are easily possible, but they typically increase the impedance of the stip-line per unit length, and would therefore require a larger number of repeater junctions in the wiring. The tradeoff between impedance per unit length and number of repeater junctions within the wiring will need to be determined for a given final circuit design. The size of a fully connected block is then roughly $L_{block} \approx P_{syn}(N_{neurons}/4) + P_{wire}(N_{neurons}/2 + \sqrt{N_{neurons}}/2)$, where L_{block} is the total length of the fully connected block of neurons, P_{syn} is the synapse pitch, $N_{neurons}$ is the number of neurons in the fully connected block, and P_{wire} is the wire pitch. This leads to an approximate size of 60 μm $\times$ 60 μm for the 16 fully connected neurons block shown in figure 8.

4. Architectures

4.1. SNNs

One of the most promising applications of superconducting electronics for neuromorphic computing is in the domain of spiking neural networks (SNNs). Biological neurons integrate signals from many synaptic and dendritic inputs and to produce a pulse when that integrated signal drives the neuron above threshold. This behavior of accumulation and discharge defines a relaxation oscillator, and relaxation oscillators are uniquely useful physical entities to serve as computational primitives. This utility derives from several contributions. To list a few, relaxation oscillators are energy efficient because they are quiescent most of the time; they are resilient to noise because their communication events are binary, all-or-nothing spikes; and they make excellent use of the time domain in that they can encode information not just in spike rate, but also in timing correlations and sequences of spike events.

The literature in this field has presented multiple designs for relaxation oscillators based on superconducting spiking neurons. At a basic level, JJs are conducive to forming spiking neurons because a JJ itself has a threshold current, and when driven above this current the junction will produce pulses of voltage and current. By combining two JJs in a superconductive loop, one forms a DC SQUID [88, 89], perhaps the most ubiquitous and useful of superconducting electronic circuits. As previously mentioned these two junctions work to balance each other, much like the Na^+ and K^+ ion channels in biological neuron membranes [3]. The DC SQUID is the primary active core of many superconducting spiking neuron designs, and neurons based on this pulsatile locus can be coaxed to demonstrate many of the dynamical responses observed in their biological counterparts [90]. Beyond neuronal response, DC SQUIDs can also be used to accomplish similar nonlinear processing and temporal filtering functions associated with dendritic processing [33].

So superconducting electronic circuits are well-equipped to perform the dynamical activities of relaxation oscillators that serve as the computational primitives for spiking neural networks. The specification that a neural network is based on spiking activity does not constrain the architectures that can be achieved. However, spiking activity is most conducive to certain types of information processing. Foremost, because spikes are temporally brief responses to stimulus, spiking neural networks are well-matched to processing stimulus that is dynamically varying. In biological systems, examples include auditory stimuli in which signals with frequency content up to tens of kilohertz are mapped onto spiking activity up to tens of hertz, as well as visual stimulus, where the persistent motions of the eyes (saccades) ensure that images on the retina are perpetually dynamic. Spiking neural networks excel at transducing these ever-changing stimuli into network activity and self-organizing the dynamics of the network so that unique stimuli become correlated with unique patterns of spiking activity.

Ensuring that unique patterns of spiking activity correlate with specific stimuli is the task of information coding, which is a primary area of neuroscientific investigation [7, 91]. The rich repertoire of responses that can be elicited from complex relaxation oscillators enable spiking neurons to encode information about a stimulus in a number of ways [92, 93]. For example, a given neuron may respond to a specific audio frequency or specific color incident on the retina by firing at a constant rate. The neuron may have a preferred audio frequency or visual color to which it responds preferentially, such that the neuron will fire most rapidly in the presence of that specific frequency or color, and its firing rate will decay the further the stimulus departs from the preferred input. Such a response is referred to as the neuron's 'tuning curve' [7]. By comparison to the responses of biological neurons, one is compelled to conjecture that superconducting neurons have extraordinary perceptual potential. While the response of the auditory system persists only to frequencies in the tens of kilohertz, the speed of JJ circuitry would appear to enable superconducting cognitive systems to directly perceive signals up to hundreds of gigahertz, endowing such systems to hear and process high-frequency microwaves. Similarly, in the visual domain, a wealth of superconducting sensors capable of detecting photons from microwaves to gamma rays offer the opportunity for superconducting neural systems to directly see this broad swath of the electromagnetic spectrum. Synaptic circuits capable of such perception with single-photon sensitivity have been discussed in section 2.2.3.

Beyond rate coding, certain neurons must respond in a manner that does not require the delay associated with integrating spikes to determine a firing rate. Individual spikes can accomplish fast response times [94]. Neurons can activate rapidly to the onset of a stimulus, sending an action potential to downstream connections which act quickly in response to a single pulse. Such a response mechanism is crucial to ensure a quick action immediately follows the onset of a stimulus that may represent danger or otherwise require prompt reaction. More complexity can be encoded and decoded in the timing between spikes produced by different neurons. Consider two neurons both transmitting signals to synapses on a third, downstream neuron. The nonlinear processing occurring in the dendritic tree of the downstream neuron enables that neuron to gain information not just about the rates of the two input neurons,

but also about the precise timing relationships between the spikes [46, 95]. If spikes from the two upstream neuron arrive at a common dendrite within a certain time window, or in a particular temporal order [46], that dendrite may pass a strong signal on to the neuron cell body, while spikes arriving with too large a temporal separation, or in the wrong order to excite that dendrite, may pass a very weak signal on to the neuron. Such intermediate processing in dendrites provide the neuron with the means to detect spike coincidences and even more elaborate spike sequences, providing the neuron with much more information about the activities of its colleagues in the network than is contained in a simple sum of synaptic activities [96].

All of these forms of information representation in spiking neurons—including rate coding, spike timing, coincidences, and sequences—can be accomplished with superconducting circuits based on the same celebrated elementary device, the DC SQUID. By judicious organization of input synapses onto a tree of SQUIDs serving as dendrites, input spike rates can be transduced to circulating DC current levels, correlated input activity can be converted to a current signal where uncorrelated activity would induce none, and specific sequences can drive a SQUID to become active, where other sequences would evoke silence. Part of the power of superconducting electronics for implementing complex spiking neurons can be attributed to this simple modularity: the DC SQUID is the cascadable building block.

Populations of neurons can function only in this modular, hierarchical manner if there is a balance between the excitation of modules responding to their preferred stimuli and the inhibition of other modules that may have a more urgent or relevant message to report. Through inhibition, various neurons and modules compete with each other for the attention of the population. Without this competition, nearly all stimulus would result in a runaway cascade of excitatory activity. With this competition, the dynamical activity perpetually shifts to allow changing stimulus to be continually reflected in the spiking activity of the network. This ever-engaged interplay between excitation and inhibition has been referred to as 'winnerless competition' [97], and it has been argued that spiking neural networks with internal competition through inhibition perform probabilistic reasoning [98] that leads ideal Bayesian inference in the presence of incomplete information [99, 100]. The hardware of superconducting electronics is naturally equipped to implement the match between excitation and inhibition. As has been discussed, coupling between synapses, dendrites, and neurons can be straightforwardly accomplished through transformers. Whether a connection is excitatory or inhibitory is simply determined by the sense of coupling in the mutual inductor used in the transformer. A given neuron can easily receive many excitatory and inhibitory connections with this technique.

4.2. Direct ANN inference in hardware

One potential architecture for superconducting neuromorphic hardware is to directly implement software style artificial neural networks (ANNs). In this type of implementation, one of the synapse styles described above can be used to weight the incoming signals. The fan-in can be performed for an arbitrarily large number inputs as long as the threshold for activating the neuron does not need to be below roughly 1% of the total input signal. However, the activation function typically used in software ANNs does not map well to single SFQ pulses with digital type fan-out and communications. While passing amplitude information in a very small network through a limited number of layers is possible, the amplitude is quickly reduced below a usable value. At this point, re-amplification of the analog value would be required in order to pass values further into the network. This type of re-amplification is typically expensive in both energy and size and therefore does not seem to be a natural fit for SFQ type neural networks. However, there are two other potential choices; one is to create a multilayer perceptron type network where the neuron acts as a threshold. The other is to use rate encoding of several pulses to convey the strength of the signal to subsequent layers.

While full scale software based neural networks, such as the transformer network [101] used in natural language processing, or resNet [102] used in image classification are too large to achieve with current JJ technology nodes, selecting smaller scale networks can provide a valuable test bed to explore the strengths and weaknesses of neuromorphic superconducting hardware. Smaller scale networks with a targeted problem could also be of more immediate practicality. For example, problems where state of the art microwave sensors are already at 4 K could use a first line of signal identification to determine which segment of the microwave band should be digitized and sent out of the cryostat for further analysis [103]. Another example is in the control and detection of q-bits that are at cryogenic temperatures. Software neural networks of a more modest size have been proven to be useful in controlling or detecting these systems and a direct hardware implementation would offer immediate advantages in speed and latency [104].

Figure 9 shows an example of an SFQ based nine pixel classification network that was simulated using WRSPICE. The network shown in figure 9(a) consists of nine input pixels which are simulated as DC to SFQ circuits that are fully connected to three different output classes. The connections are weighted by 27 MJJ style inductive dividers and the fan-in circuits are flux based as described in [54]. The nine pixel classification problem was originally simulated using a sigmoid based neural network in python to determine the weights. The weights were then linearly mapped to the critical current values of the simulated MJJs. Figure 9(b) shows the output of the 3 postsynaptic JJ neurons, each of which had a fan-in of nine weighted inputs. The output is the simulated phase of one of the output JJ in a JJ neuron configuration described above. 2π steps can be seen indicating that the output JJ neuron fired an SFQ pulse, which can be used as the output of the circuit. Note that the circuit is running at a 100 GHz clock demonstrating one of the potential advantages of SFQ type neuromorphic computing, namely, very high speed. The initial latency of the circuit is about 30 ps, which is the time required for the SFQ pulse to travel through the network of JJs. However, one need not wait for the output before sending

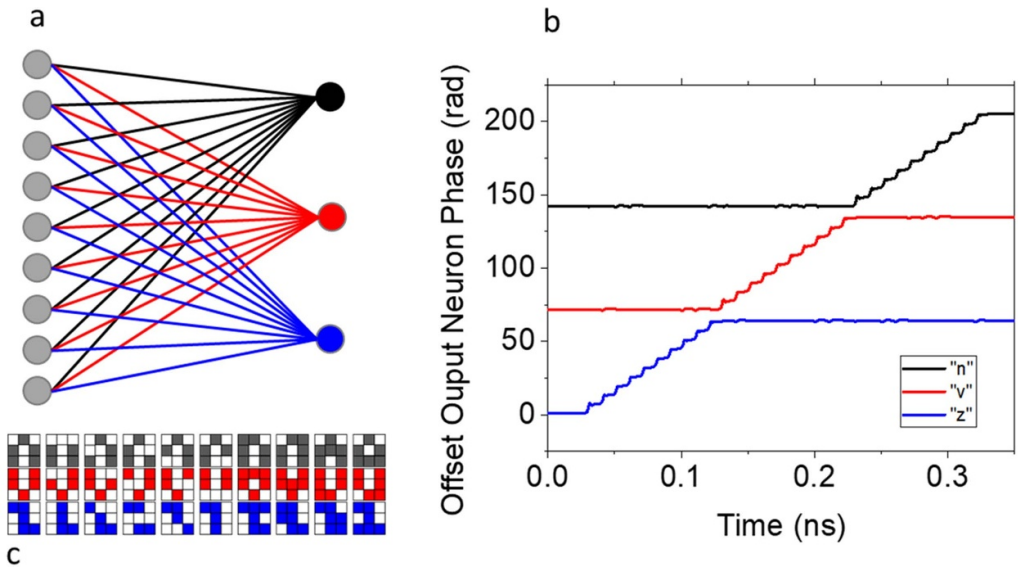

Figure 9. (a) Schematic of 9 pixel JJ classifier network that was simulated using WRSPICE. Each of the lines includes an adjustable strength synaptic element. (b) Phase of each of the output neuron junctions offset for clarity. 2π steps indicate one SFQ pulse was fired as a result of the input image. (c) The input images representing n, v, and z and allowing for any one pixel of 'noise' on the input image.

the next input. Thus, such a network could be fed new images every 10 ps. In general, as the SFQ type ANNs grow in size the latency will increase, but with careful design 100 GHz signal input rates should be within current technological capabilities. Figure 9(c) shows the 30 test cases; 10 each from one of three classes representing the letters *n*, *v*, or *z* using nine pixels. One of the pixels of any of these letters was then turned on or off to represent noise. This example problem was first shown for a small memristor hardware network [105].

4.3. Biologically realistic networks

The typical network structure for both ANNs and SNNs is feed-forward, where the input stimuli couple to a set of inner layers of neurons which are ordered, i.e. the first layer couples to the second, the second layer couples to the third, and so forth. The so-called recurrent connections, where for example the second layer couples back to the first, are usually avoided, because they make the training more difficult. However, recurrent connections occur frequently in mammalian brains. Given the brain's extraordinary information processing capability and energy efficiency, it is tempting to study network structures with recurrent connections and asses their capability in neuromorphic hardware.

A simple example of a recurrent connection occurs in two mutually-coupled spiking neurons, where the first neuron is coupled to the second and the second is coupled back to the first. The mutual coupling occurs through two synapses and can include a time delay along the axons and dendrites. Mutually-coupled neurons tend to synchronize, with the two neurons settling into a long-term firing state where they both spike at the same rate with a fixed phase relationship. If the neurons and synapses are identical, only two possible phase relationships exist: in-phase, where the neurons fire at exactly the same time, or anti-phase, where they fire 180 degrees out of phase with each other. The synaptic strength and the time delay determine whether the synchronization is in-phase or anti-phase, in a complicated nonlinear relationship which depends on the dynamics of the neurons.

If the synaptic strength or the time delay of a connection slowly varies, a synchronized system can toggle from in-phase to anti-phase or vice-versa in what is known as a phase-flip bifurcation. Phase-flip bifurcations are a generic feature of time-coupled nonlinear oscillators and have been studied extensively in the bifurcation community. They have also been observed in mammalian brains, where they are quite common; in Dotson and Gray [106], the authors recorded time traces from 16 locations in a primate brain and found phase-flip bifurcations in 9 of them. Although the details of how a phase-flip bifurcation contributes to the brain's higher functioning is still under study, it is well known that oscillator phase is used to encode information, so it is likely that its role is significant.

In Segall *et al* [5], a mutually coupled system of two JJ neurons was studied. Josephson transmission lines were used as axons and SQUIDs were used as synapses. Figure 10(a) shows a scanning electron micrograph of the circuit. The neurons were biased to fire continuously, and the flux and bias current to the SQUID were varied to change the coupling strength and the delay. SFQ-to-DC converters were used to measure the firing frequency of both neurons, and an SFQ merger (OR-gate) was used to measure the phase relationship. In the experiments, a large region of parameter space showed synchronization of both in-phase and anti-phase types. Phase-flip bifurcations were found both as a function of the SQUID flux and bias current. Figure 10(b) shows the bifurcation diagram of the system, with red and blue regions indicating in-phase and anti-phase states.

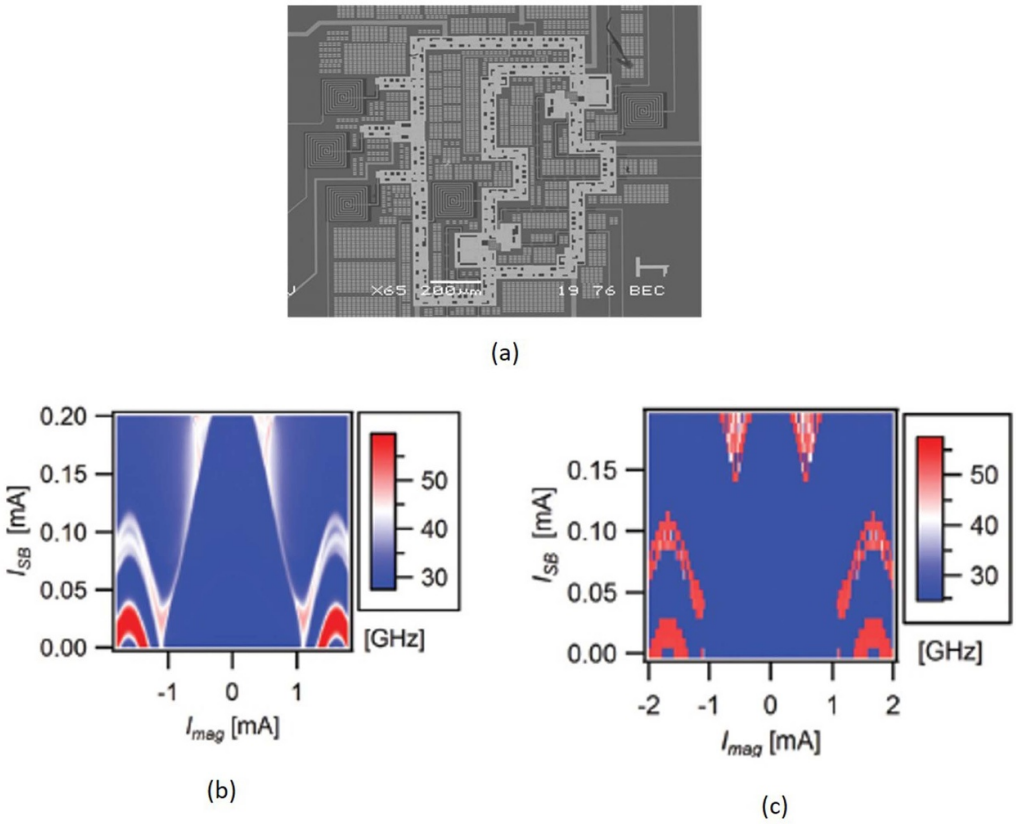

Figure 10. Phase flip bifurcation as described in Segall *et al* [5]. (a) Circuit picture of two mutually-coupled JJ neurons, with axons, synapses and readout circuitry. (b) Measured bifurcation diagram showing in-phase (blue) and anti-phase (red) regions. The *x*- and *y*-axis are control currents for the synapse. (c) WR-SPICE simulation of the same parameter space as in (b).

The circuit studied by Segall *et al* [5] was also simulated using the WR-SPICE simulation package for superconducting electronics. Figure 10(c) shows the simulated data from (b) and very good agreement was found. The simulation times were over four orders of magnitude longer than the measurement times. In future, larger networks that ratio would grow even larger, dis-proportionally with the number of neurons. This leads one to consider efficient brain simulation as a possible application of superconducting neuromorphic hardware.

4.4. Winner take all

Neuromorphic hardware can also be organized into bio-inspired architectures for testing theories of how biological neurons interact. For example, the winner-takes-all (WTA) framework is a computational model that describes how groups of neurons develop selectivity through competition [107]. While many models have been proposed, figure 11(a) highlights one WTA network developed for spiking neurons in which two shared inhibitors create competition between the outputs, suppressing them until only one (the winner) is left firing [108]. As illustrated in figure 11, these networks are highly interconnected; each output neuron has two inhibitory input connections and two excitatory connections (one from its respective input neuron and one from its own output). In this particular case, the inhibitors are excited by the output neurons and play distinct roles based on their different firing thresholds. The convergence inhibitor Z_c creates competition and fires as long as two outputs are active, which can be achieved by setting its threshold appropriately. The stability inhibitor Z_s has a lower firing threshold so that it fires as long as just one output is active. Its primary responsibility is keeping the losing neurons suppressed, thereby saving the result of the competition.

A key feature of this particular design is that the output neurons are identical and stochastic, meaning that the winner is determined by probability. This aspect is distinct from other WTA models that use neurons with different firing frequencies. Since stochasticity is frequently observed in biological neurons, WTA models that explore how stochastic neurons develop selective pathways may be powerful tools for understanding large-scale dynamics in the human brain.

Recent work demonstrated how the stochastically varying firing thresholds of superconducting nanowire-based neurons can be used for simulating these types of WTA networks [18]. Superconducting nanowires are susceptible to thermal and quantum fluctuations, causing them to switch prematurely at currents below their critical current [109]. As a result, their measured switching currents form a probabilistic distribution. By incorporating these probabilistic switching events into a circuit-based nanowire neuron model, it was possible to simulate a WTA competition between three nanowire neurons and demonstrate that the winner changes from iteration to

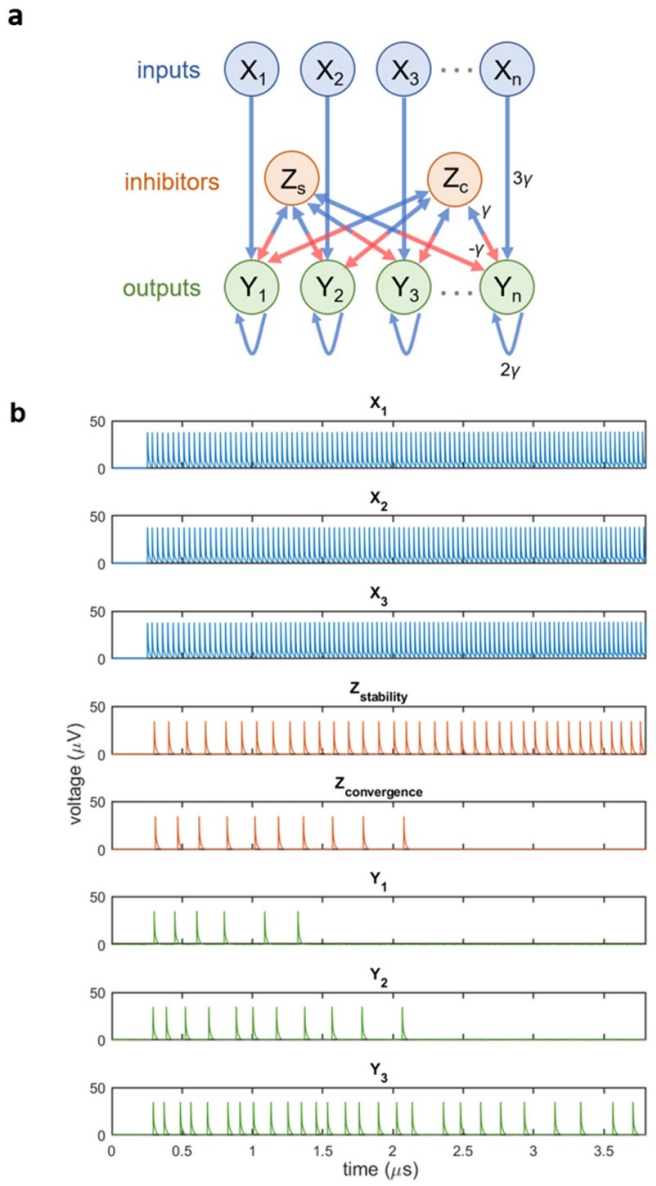

Figure 11. (a) Schematic of a two-inhibitor WTA network. Synaptic weights are indicated relative to the dimensionless parameter γ. (b) A simulated WTA competition using nanowire spiking neurons. In this particular competition, the third output neuron was the winner. Reprinted with permission from [18]. © 2020, American Chemical Society.

iteration as a result of probability. This example emphasizes how intrinsic device stochasticity–usually avoided in digital circuits–can instead be harnessed in neuromorphic applications to simulate theories of how populations of biological neurons interact.

4.5. Mixed analog digital

Digital and analog components can be used to build mixed-signal spiking neuromorphic architectures that combine the benefits of both types of circuits while avoiding some of their disadvantages. In particular, the mixed-signal approach typically leverages low-noise digital communication to build large networks of complex analog neurons. This mixed-signal approach has been explored by several semiconductor neuromorphic architectures [82–84] as a way to build systems with millions of complex neurons and billions of synapses. Many of the same ideas can be incorporated into superconducting neuromorphic systems by using superconducting digital logic to build some aspects of the architecture such as the communication network between neurons.

Some of the more prominent semiconductor spiking neuromorphic architectures [85, 86] utilize only digital components because this avoids the challenges and overheads that arise when interfacing digital and analog components. However, these digital approaches require many more circuit elements in order to approximate the behavior of inherently analog biological neurons. As a result, most digital architectures, such as TrueNorth and Loihi, settle for the Leaky-Integrate-and-Fire (LIF) model of the soma, which is less biologically suggestive but less computationally complex than other models [110]. Analog approaches, however, are typically able to efficiently implement more biologically relevant models of the soma such as the Adaptive Exponential Integrate and Fire model or the Adaptive Quadratic Integrate and Fire model. These soma models capture more of the behaviors of the biological neuron and could provide important functionality for more advanced neural networks. This functionality is desirable but implementing it in a digital system often prohibitively limits the number of neurons that can fit on a chip.

Similarly, large-scale semiconductor spiking neuromorphic systems are typically difficult to implement using purely analog components due to the noise that is introduced by those components. In addition, in order to scale efficiently, physical components in the system often need to be shared between different neurons in a time multiplexed fashion. This avoids the problem of overprovisioning the system when only a small number of components will be in use at any one time relative to the total number of neurons being simulated. It also avoids the need to provide enough wiring layers to connect thousands or millions of neurons together. There is a limit to the number of wiring layers that can reasonably be provided by current fabrication processes and this limit in turn limits the total number of neurons that can be directly interconnected. Sharing wires and other components in a time-multiplexed fashion can address these problems but such a time-multiplexed arrangement is much more difficult to implement in an analog architecture than it is in a digital one.

The situation is slightly different for superconducting electronics, but the overall principle still holds that using a mixed-signal architecture enables the system to simultaneously support complex neuron models and scalability. Superconducting digital logic enables the time-multiplexing of components in the architecture in a similar manner to some semiconductor approaches. This allows for large numbers of virtual neurons to be simulated by sharing hardware and wires throughout the system at the cost of some performance. At the same time, superconducting circuits can be designed that implement complex neuron models using relatively few circuit components [3]. These compact soma circuits enable the architecture to incorporate a soma model that closely

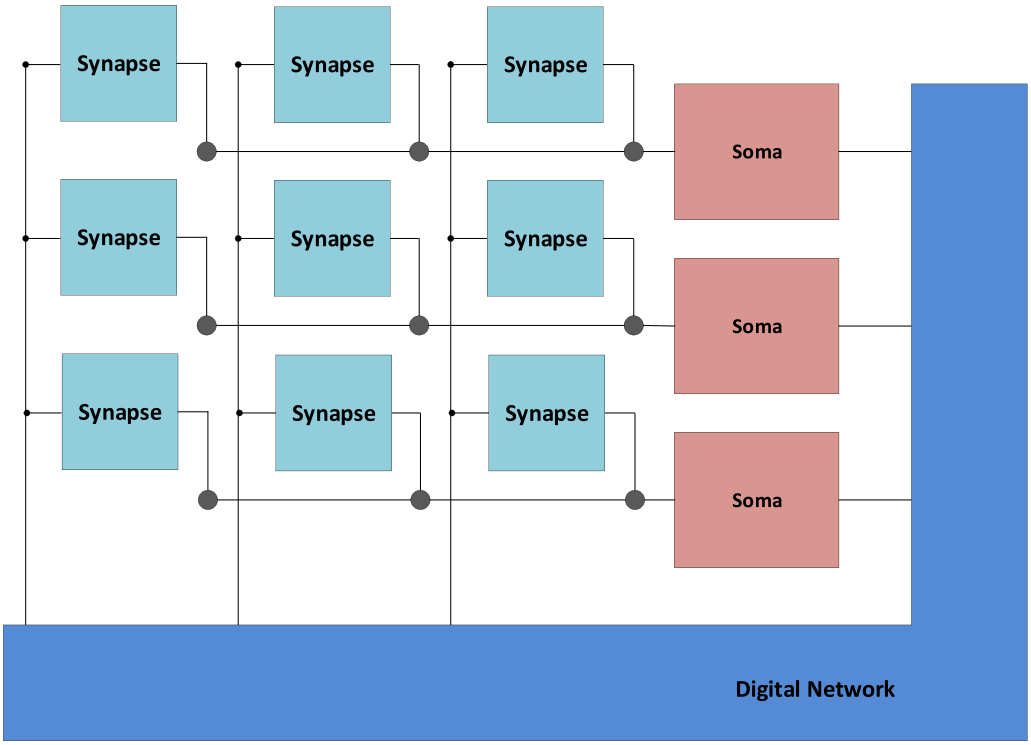

Figure 12. The shared axon neuromorphic architecture, which shares physical connections between neurons in an effort to reduce the wiring density of large-scale networks.

resembles the Hodgkin–Huxely model with much less hardware than would be required in a semiconductor implementation. By combining the time-multiplexed digital components with the compact soma circuits it should be possible to build a neuromorphic system that captures biologically suggestive neuron behaviors and interactions at a large scale. In addition, the SFQ pulses used by the digital logic are the same pulses that are produced by the soma circuits. This allows for much more straightforward interfacing between the two types of circuits without the need for an ADC. In some architectures a DAC may still be required to translate from a digital output into the analog input needed by some soma circuits. Finally, because the superconducting digital circuits tend to run at much higher clock rates than their semiconductor counterparts, the biologically suggestive simulation can take place at time scales that are much faster than biological real time.

Directly connecting neurons together is the most straightforward way to build neuromorphic systems but the number of neurons that can be implemented in this fashion is ultimately limited by wiring and circuit density. To address this, several different schemes exist that share different degrees of the hardware between virtual neurons in neuromorphic architectures such as the shared axon and shared synapse approaches, shown in figures 12 and 13 respectively. In the shared axon approach, the wires that are used to connect the neurons are shared by using a digital network to route packets between the different virtual neurons. In this way, the number of wires needed to implement a very large fully connected network can be greatly reduced. To implement this digital network, the spiking events are typically encoded as Address event representation (AER) packets which communicate the timing and location of each spike to the appropriate post synaptic neurons [61–63]. By capturing the timing information of each spike in the packet the relationships between the spike timings of different neurons in the system is preserved even though a digital network is used to communicate the spikes. The synapse connections between neurons are recorded in the routing tables of the digital network which are used to determine where to send each AER packet. Additional information about the different neuromorphic digital networks and routing approaches can be found in [111]. Importantly, this digital network approach allows for arbitrary connectivity between neurons in the simulation because the connections between virtual neurons are just entries in the table of a router. As a result, the same hardware can be reused to implement many different neural networks and perform different experiments.

The shared synapse approach builds upon the shared axon approach by also sharing the same synapse memory arrays between neurons. This allows synapse weights for different neurons in the network to be stored in the same memory subsystem which can improve utilization and reduce the area requirements of the memory hardware. In addition, this approach allows for the same weight value to be used by multiple virtual neurons thereby reducing the total amount of synapse weight memory needed for a network. Sharing synapse weights between virtual neurons is generally not compatible with on-line learning as the values of individual synapses need to be able to change independently to support that functionality. More aggressive approaches have also been explored in an

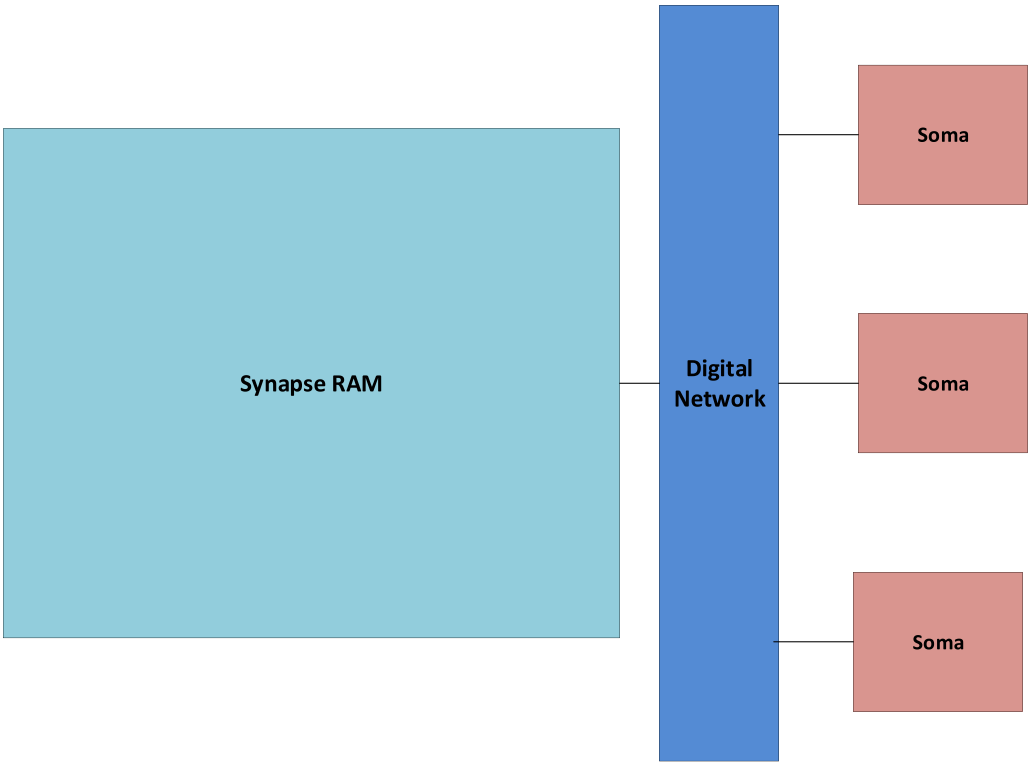

Figure 13. The shared synapse neuromorphic architecture, which shares the physical memories used to store synapse weights between neurons in an effort to reduce the memory overheads required for large-scale networks.

effort to further improve scalability, such as the shared dendrite approach used by Neurogrid [83].

The shared axon and shared synapse architectures that are enabled by the mixed-signal approach address two of the more significant challenges facing scalable superconducting neuromorphic systems. In particular, the number of directly connected physical neurons that can be implemented on a single die is significantly less in the superconducting case. This is because the circuit and wiring density of superconducting electronics is typically much less than that of semiconductors. By implementing a shared axon architecture, the wiring density limitation can be greatly alleviated because the wires are shared between virtual neurons. Similarly, the shared synapse approach can be used to alleviate some of the scaling limitations that are imposed by the limited density of superconducting memory technologies. Additional digital components could also potentially be shared by time multiplexing them to further improve the scalability of superconducting neuromorphic systems. As a result, the mixed signal approach allows superconducting neuromorphic systems to scale to much larger numbers of neurons that might otherwise be possible given the current limitations of the technology. An example mixed-signal architecture is depicted in figure 14.

By combining large numbers of biologically suggestive neurons into a simulation that runs at speeds much faster than biological real time, mixed signal superconducting neuromorphic systems have the potential to enable new investigations in the area of computational neuroscience. Simulating large numbers of biologically suggestive neurons in software is a computationally demanding task that results in prohibitively long simulation times. Supporting these sorts of simulations in superconducting hardware would enable experiments that previously were not feasible. In particular, experiments involving the interactions between large numbers of biologically suggestive neurons could be a valuable tool in the effort to understand how the biological brain works. Ultimately, these experiments and the superconducting hardware that supports them could provide critical insights in the push to general artificial intelligence.

4.6. Reservoir computing

The superconducting neuromorphic architectures described above all seek an operational mode evocative of biological neural networks, even though they differ in the particulars of their physical implementation. Reservoir computing (RC) offers the remarkable possibility of subsuming the dynamics of any one of these physical substrates in order to perform computation in a different operational mode that offers a simpler training procedure and eliminates the need for storing most synaptic weights. In order to understand how superconducting circuits can function as a computational resource in this manner, it is helpful to explain the software neural-network origins of the RC paradigm.

RC emerged as an alternative training methodology for recurrent neural networks (RNNs) that are difficult to train using traditional back-propagation methods [112, 113]. Attempts to train complex (i.e. large and highly connected) RNNs mainly resulted in changes to the output weights

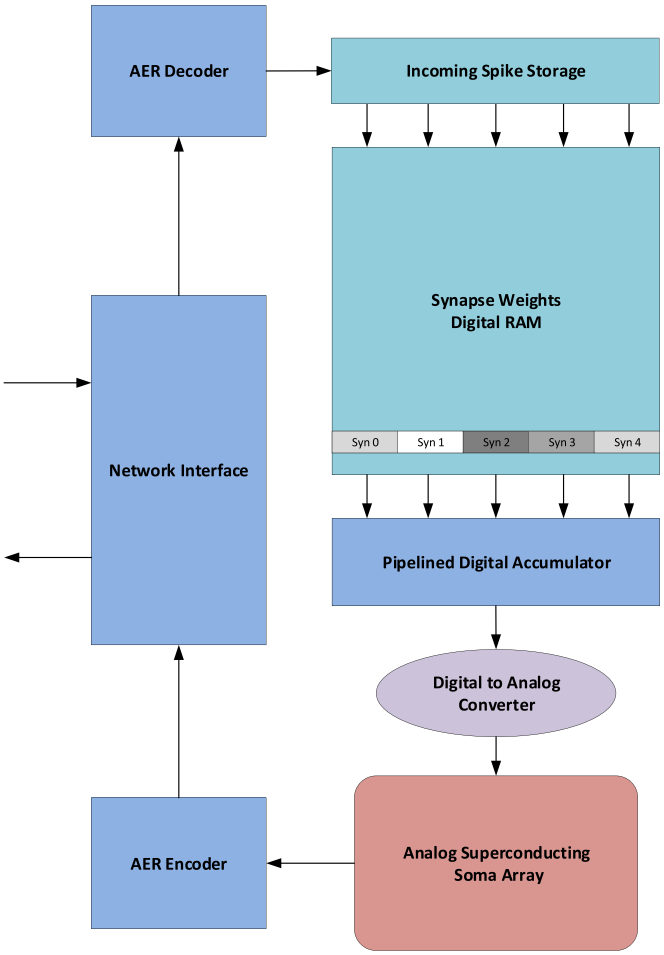

Figure 14. An example mixed-signal architecture with each of the major components shown.

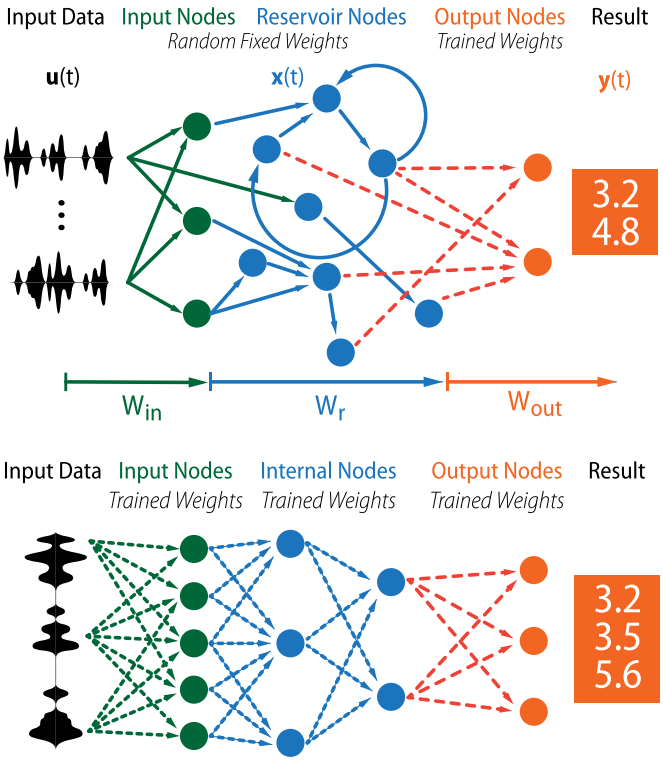

Figure 15. Comparison between an echo state network (ESN) and an artificial neural network (ANN). (a) The ESN processes time-domain input data and has fixed weights apart from its outputs. The ESN's construction otherwise mirrors a recursive neural network (RNN). (b) The ANN operates on parallel input data, and all of its connections are based on trainable weights.

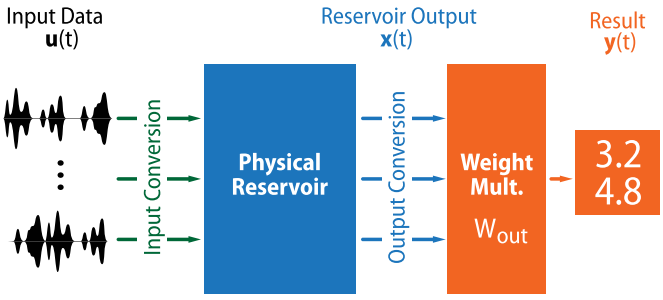

Figure 16. A block diagram of a physical reservoir supplanting the ESN. The only explicit operations performed outside of the reservoir are input/output encoding and output weight multiplication.

W_{out} of the system, while the internal weights W_r remained largely unchanged. Embracing this insensitivity as a tenet, RC assumes that W_r are initialized randomly and fixed; no attempts are made to train them. In most cases the input weights W_{in} are treated in the same manner. These RNN variants are called echo state networks (ESNs), an example of which is shown in figure 15 in comparison to a simple ANN architecture in which every single weight must be trained to solve a particular problem. ESNs typically employ supervised learning and are trained by finding the output weights W_{out} that produce the best estimates $\mathbf{y}(t) = W_{\text{out}}\mathbf{x}(t)$ of the ground truth training data $\tilde{\mathbf{y}}(t)$ for input data $\mathbf{u}(t)$ and internal reservoir state $\mathbf{x}(t)$. The simplest approach to this training is to use the Moore–Penrose pseudo-inverse: $W_{\text{out}} = \tilde{\mathbf{y}}(t)\mathbf{x}(t)^{\dagger}$ which gives the solution explicitly and without the need for iterative back-propagation. A δ update rule can also be used that performs single-step gradient descent on the linear outputs [114], continually tuning weights in response to changes in input data conditions or even the reservoir itself. No matter the particular training methodology the reservoir may be used to solve multiple problems, since W_{out} is not integral to the reservoir. The recurrent nature of ESNs (and RC more generally) makes them well-suited to time-domain problems, and so we have written these quantities as a function of time t, though in the case of ESNs this is understood to mean a discrete-time basis where any time dependent quantity $\mathbf{a}(t)$ is taken to imply the series of points $[a_0, a_1, \ldots, a_n]$.

Unfortunately, attempts to directly accelerate ESN execution on a hardware platform are ultimately subject to the same computational and memory bandwidth constraints that have motivated much of the earlier work in this review. For RC, however, another opportunity emerges: the black-box nature of the RC and the lack of a need for controlling its internal weights means that it can be replaced by a physical system whose dynamics form a substrate for so-called physical reservoir computing (PRC) [115], as represented pictorially in figure 16. So long as this physical system exhibits

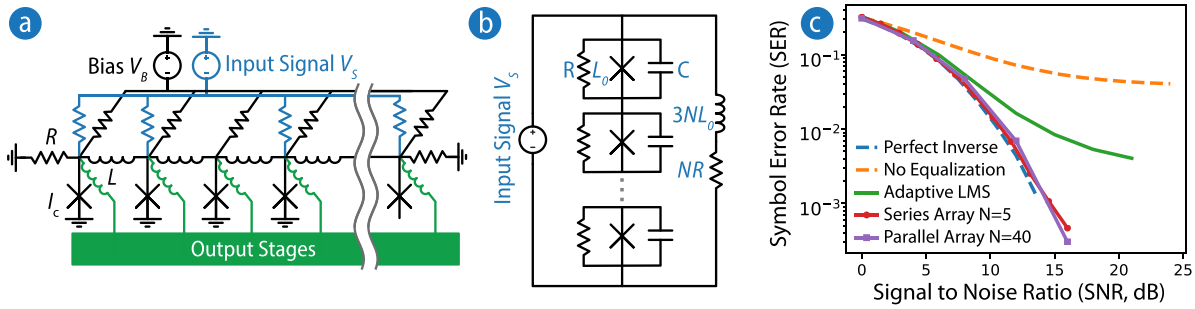

Figure 17. Two potential reservoir computing architectures based on JJs. (a) A 'parallel' array of junctions analogous to a Josephson transmission line (JTL) and (b) a series array where JJs are coupled via a global load. (c) The performance of both of these architectures where performing channel equalization, as compared to an adaptive least mean squares (LMS) technique and to the cases where the exact channel inverse is known and no equalization is performed.

sufficiently high dimensionality, non-linearity, and the fading-memory property (i.e. it has a short-term dependence on the reservoir inputs) it can be exploited for this purpose. Conveniently, there is no need to explicitly store the internal and input weights W_{r} and W_{in} since they are provided naturally by the design of the reservoir. The emergence of the PRC paradigm unsurprisingly resulted in an explosion of different hardware reservoir realizations—optical, spintronic, mechanical, electronic, etc—whose diversity speaks to the broad applicability of PRC [116]. Here we summarize the recently introduced implementation of RC on a superconducting circuit platform [117], which provides considerable advantages in terms of speed and energy efficiency compared to other PRC embodiments.

Starting with simple circuits such as the Josephson transmission line (JTL) in figure 17(a) or the globally coupled series array in figure 17(b), one must first choose input and output encodings that facilitate information processing by the reservoir. In order that the JJ voltages are easily measured, it is convenient to operate these circuits above the critical current I_c of the JJs, at which point they enter a complex coupled oscillatory state wherein all of the junctions undergo relaxation oscillations. In the JTL these are soliton dynamics, as the circuit equations mirror the Sine–Gordon equation [118]. In the series array these dynamics are strongly coupled but kept away from the precipice of synchrony that is readily observed in such structures [119]. For these PRC embodiments, the inputs are raw waveforms applied as currents without pre-processing or encoding. The reservoir's dynamics are interpreted through the lens of a multi-pulse encoding of the dynamics in some fraction of the JJs in the reservoir. The spiking rates give similar information as the time-average voltages, either of which can be used for the purposes of reservoir computing depending on which is experimentally more convenient. The efficacy of these reservoirs for signal processing is shown in figure 17(c), where the reservoirs are asked to perform channel equalization on a 4-level amplitude modulated signal subject to multipath-fading, nonlinear distortion, and additive white Gaussian noise (AWGN). Remarkably both the series and parallel (JTL) arrays are able to perform at the level of the 'exact' channel inverse, which of course cannot invert the AWGN. Both superconducting circuits easily outperform a traditional channel equalization technique, adaptive least mean squares (LMS), which struggles with the nonlinear distortion effects of the channel. This performance is highly encouraging, especially in light of the minimal number of JJs required: 40 in the case of the JTL and 5 in the case of the series array. Also impressive are the data rates: 10 Gb s^{-1} to 20 Gb s^{-1} for this signal equalization task with the possibility of scaling to 100 Gb s^{-1}. In [117] a readout chain based on simple RSFQ circuits was considered, which converts the analog oscillations of the reservoir JJs into decimated digital signals that can be brought to room temperature for direct measurement or passed to additional RSFQ circuitry that can perform the final weight multiplication. While multiplication can be performed quickly enough in RSFQ to support these high data rates [120, 121], it comes at the cost of a nearly thousand-fold increase in the required junction count.

Despite the allure of interfacing these superconducting reservoirs directly to a compatible logic family, a more enticing possibilities is combining superconducting reservoirs with the technologies mentioned earlier in this review, bringing to bear the power and flexibility of weight-based JJ computing in a targeted fashion. The most straightforward combination of JJ-based RCs and neural networks would have the latter serve as a readout layer for the reservoir. This scheme has been demonstrated in liquid state machines—effectively spiking implementations of reservoirs—where a trained perceptron output layer takes the place of an output weight matrix [122]. This could eliminate the need for explicit multiplication using SFQ or other logic implementations and further reduce the junction counts in a hybrid architecture. Additional extensions of a hybrid architecture could mirror the properties of so-called Deep-ESNs [123] which combine multiple reservoirs with intermediate trained output layers, or by adding auto-encoding stages that enhance the generalization capabilities of the system.

Overall, superconducting RC provides a powerful tool in the superconducting neuromorphic toolkit that sidesteps the traditional constraints of local memory resources while still serving as a general purpose computational architecture whose functionality is determined solely by the output weights of the system. The extremely high oscillation rates of JJs allow for inference at extremely high speeds (to 100 Gb s^{-1} and

potentially beyond), while maintaining full compatibility with other digital and neuromorphic superconducting platforms.

5. Summary

We presented an overview of the current state of neuromorphic computing based on superconducting electronics. JJs and superconducting nanowires can implement bio-inspired functionality directly at the device level. For example, JJs exhibit naturally spiking behavior analogous to the action potential that neurons in the brain exhibit. Small scale circuits with only a few devices can extend this functionality dramatically depending on the desired properties. For example, two JJs can mimic the ion channel dynamics that are observed in firing neurons, and a nanowire resistor combination can mimic the relaxation oscillations observed in the brain when several neurons are connected. These device properties combined with superconducting communications, which work well with time domain spiking signals, provide a diverse set of bio-inspired primitives for performing computation.

The way in which these devices or small circuits are connected depends significantly on the target application. We discuss some of the options including small-scale fully connected analog networks and reservoirs. These small analog blocks can be further connected with digital communications to form more powerful networks. Such mixed analog digital superconducting neuromorphic systems have great promise for their ability to implement more general computational functionality. In addition to all electrical connectivity, one can also integrate optical communications. The integration of photonics with superconducting electronics may be particularly effective because of highly sensitive cryogenic single photon detectors, which reduce the requirements on integrated light sources, and on the light sources themselves that may benefit from cryogenic operation.

The fabrication of superconducting neuromorphic processors that can approach the size and scope of current day software based neural networks seems difficult, to say the least. The current state-of-the-art in digital JJ circuits is still at a relatively low integration density compared to CMOS. In part, this is due to the stringent requirements of digital computation, which is not tolerant of faults even at a very low level. Superconducting neuromorphic computing is approximate in nature and therefore has the potential for acceptable operation even in the presences of low-level errors from sources such as flux trapping. Whether or not this potential advantage enables greater integration density for superconducting neuromorphic systems remains to be seen. However, targeted applications such as microwave signal sensing or quantum state readout may be far more within reach and could drive the field forward in the near term. The speed and energy advantages of superconducting neuromorphic computing, particularly for applications that already operate in a cryogenic environment, are significant. This combined with the ability to leverage digital superconducting fabrication techniques make superconducting neuromorphic computing a very appealing area of research and development.

Data availability statement

The data that support the findings of this study are available upon reasonable request from the authors.

Disclaimer

This document does not contain technology or technical data controlled under either the U.S. International Traffic in Arms Regulations or the U.S. Export Administration Regulations.

ORCID iDs

Michael Schneider https://orcid.org/0000-0003-1928-6662
Jeff Shainline https://orcid.org/0000-0002-6102-5880

References

[1] Ishida K *et al* 2020 SuperNPU: an extremely fast neural processing unit using superconducting logic devices *2020 53rd Annual IEEE/ACM Int. Symp. on Microarchitecture (MICRO)* (IEEE) pp 58–72
[2] Ishida K *et al* 2021 Superconductor computing for neural networks *IEEE Micro* **41** 19–26
[3] Crotty P, Schult D and Segall K 2010 Josephson junction simulation of neurons *Phys. Rev.* E **82** 011914
[4] Likharev K K and Semenov V K 1991 RSFQ logic/memory family: a new Josephson-junction technology for sub-terahertz-clock-frequency digital systems *IEEE Trans. Appl. Supercond.* **1** 3–28
[5] Segall K, LeGro M, Kaplan S, Svitelskiy O, Khadka S, Crotty P and Schult D 2017 Synchronization dynamics on the picosecond time scale in coupled Josephson junction neurons *Phys. Rev.* E **95** 032220
[6] Schneider M L, Donnelly C A, Russek S E, Baek B, Pufall M R, Hopkins P F, Dresselhaus P D, Benz S P and Rippard W H 2018 Ultralow power artificial synapses using nanotextured magnetic Josephson junctions *Sci. Adv.* **4** e1701329
[7] Dayan P and Abbott L F 2001 *Theoretical Neuroscience* (Cambridge, MA: MIT Press)
[8] Tolpygo S K and Semenov V K 2020 Increasing integration scale of superconductor electronics beyond one million Josephson junctions *J. Phys.: Conf. Ser.* **1559** 012002
[9] Scott Holmes D, Ripple A L and Manheimer M A 2013 Energy-efficient superconducting computing—power budgets and requirements *IEEE Trans. Appl. Supercond.* **23** 1701610
[10] Furber S and Temple S 2007 Neural systems engineering *J. R. Soc. Interface* **4** 193–206
[11] Yamanashi Y, Umeda K and Yoshikawa N 2013 Pseudo sigmoid function generator for a superconductive neural network *IEEE Trans. Appl. Supercond.* **23** 1701004
[12] Chiarello F, Carelli P, Castellano M G and Torrioli G 2013 Artificial neural network based on SQUIDs: demonstration of network training and operation *Supercond. Sci. Technol.* **26** 125009
[13] Schegolev A E, Klenov N V, Soloviev I I and Tereshonok M V 2016 Adiabatic superconducting cells for ultra-low-power artificial neural networks *Beilstein J. Nanotechnol.* **7** 1397–403
[14] Segall K, Guo S, Crotty P, Schult D and Miller M 2014 Phase-flip bifurcation in a coupled Josephson junction neuron system *Physica* B **455** 71–75

[15] Bozbey A, Karamuftuoglu M A, Razmkhah S and Ozbayoglu M 2018 Single flux quantum based ultrahigh speed spiking neuromorphic processor architecture (arXiv:1812.10354)

[16] Cheng R, Goteti U S and Hamilton M C 2018 Spiking neuron circuits using superconducting quantum phase-slip junctions *J. Appl. Phys.* **124** 152126

[17] Toomey E, Segall K and Berggren K K 2019 Design of a power efficient artificial neuron using superconducting nanowires *Front. Neurosci.* **13** 933

[18] Toomey E, Segall K, Castellani M, Colangelo M, Lynch N and Berggren K K 2020 Superconducting nanowire spiking element for neural networks *Nano Lett.* **20** 8059–66

[19] Hebb D O 1949 *The Organization of Behavior: A Neuropsycholocigal Theory* (*A Wiley Book in Clinical Psychology* vol 62) (New York: Wiley) p 78

[20] Rumelhart D E, Hinton G E and Williams R J 1986 Learning representations by back-propagating errors *Nature* **323** 533–6

[21] Klenov N V, Schegolev A E, Soloviev I I, Bakurskiy S V and Tereshonok M V 2018 Energy efficient superconducting neural networks for high-speed intellectual data processing systems *IEEE Trans. Appl. Supercond.* **28** 1301006

[22] Buzdin A I 2005 Proximity effects in superconductor-ferromagnet heterostructures *Rev. Mod. Phys.* **77** 935

[23] Bergeret F S, Volkov A F and Efetov K B 2005 Odd triplet superconductivity and related phenomena in superconductor-ferromagnet structures *Rev. Mod. Phys.* **77** 1321

[24] Linder J and Robinson J W A 2015 Superconducting spintronics *Nat. Phys.* **11** 307–15

[25] Feofanov A K *et al* 2010 Implementation of superconductor/ferromagnet/superconductor π-shifters in superconducting digital and quantum circuits *Nat. Phys.* **6** 593–7

[26] Khaire T S, Khasawneh M A, Pratt W P Jr and Birge N O 2010 Observation of spin-triplet superconductivity in co-based Josephson junctions *Phys. Rev. Lett.* **104** 137002

[27] Bell C, Burnell G, Leung C W, Tarte E J, Kang D-J and Blamire M G 2004 Controllable Josephson current through a pseudospin-valve structure *Appl. Phys. Lett.* **84** 1153–5

[28] Van Duzer T and Turner C W 1981 *Principles of Superconductive Devices and Circuits* (Englewood Cliffs, NJ: Prentice-Hall)

[29] Soloviev I I, Schegolev A E, Klenov N V, Bakurskiy S V, Kupriyanov M Y, Tereshonok M V, Shadrin A V, Stolyarov V S and Golubov A A 2018 Adiabatic superconducting artificial neural network: basic cells *J. Appl. Phys.* **124** 152113

[30] Primavera B A and Shainline J M 2021 Considerations for neuromorphic supercomputing in semiconducting and superconducting optoelectronic hardware *Front. Neurosci.* **15** 732368

[31] Shainline J M, Buckley S M, McCaughan A N, Chiles J, Jafari-Salim A, Mirin R P and Nam S W 2018 Circuit designs for superconducting optoelectronic loop neurons *J. Appl. Phys.* **124** 152130

[32] Shainline J M *et al* 2019 Superconducting optoelectronic loop neurons *J. Appl. Phys.* **126** 044902

[33] Shainline J M 2020 Fluxonic processing of photonic synapse events *IEEE J. Sel. Top. Quantum Electron.* **26** 7700315

[34] Onen M, Turchetti M, Butters B A, Bionta M R, Keathley P D and Berggren K K 2020 Single-photon single-flux coupled detectors *Nano Lett.* **20** 664

[35] Toomey E, Onen M, Colangelo M, Butters B A, McCaughan A N and Berggren K K 2019 Bridging the gap between nanowires and Josephson junctions: a superconducting device based on controlled fluxon transfer *Phys. Rev. Appl.* **11** 034006

[36] Primavera B A and Shainline J M 2021 An active dendritic tree can mitigate fan-in limitations in superconducting neurons (arXiv:2107.05777)

[37] Gidon A, Zolnik T A, Fidzinski P, Bolduan F, Papoutsi A, Poirazi P, Holtkamp M, Vida I and Larkum M E 2020 Dendritic action potentials and computation in human layer 2/3 cortical neurons *Science* **367** 83–87

[38] Beniaguev D, Segev I and London M 2021 Single cortical neurons as deep artificial neural networks *Neuron* **109** 2727–39

[39] London M and Häusser M 2005 Dendritic Computation *Annu. Rev. Neurosci.* **28** 503

[40] Schegolev A, Klenov N, Soloviev I and Tereshonok M 2020 Learning cell for superconducting neural networks *Supercond. Sci. Technol.* **34** 015006

[41] Riedmiller M and Braun H 1993 A direct adaptive method for faster backpropagation learning: the RPROP algorithm *IEEE Int. Conf. Neural Networks* (IEEE) pp 586–91

[42] Mizugaki Y, Nakajima K, Sawada Y and Yamashita T 1993 Superconducting neural circuits using fluxon pulses *Appl. Phys. Lett.* **62** 762–4

[43] Onomi T and Nakajima K 2014 An improved superconducting neural circuit and its application for a neural network solving a combinatorial optimization problem *J. Phys.: Conf. Ser.* **507** 042029

[44] Cheng R, Goteti U S and Hamilton M C 2019 Superconducting neuromorphic computing using quantum phase-slip junctions *IEEE Trans. Appl. Supercond.* **29** 1300505

[45] Stuart G J and Spruston N 2015 Dendritic integration: 60 years of progress *Nat. Neurosci.* **18** 1713

[46] Hawkins J and Ahmad S 2016 Why neurons have thousands of synapses, a theory of sequence memory in neocortex *Front. Neural Circuits* **10** 23

[47] Sardi S, Vardi R, Sheinin A, Goldental A and Kanter I 2017 New types of experiments reveal that a neuron functions as multiple independent threshold units *Sci. Rep.* **7** 18036

[48] Mel B W, Schiller J and Poirazi P 2017 Synaptic plasticity in dendrites: complications and coping strategies *Curr. Opin. Neurobiol.* **43** 177

[49] Jesper Sjöström P, Rancz E A, Roth A and Häusser M 2008 Dendritic excitability and synaptic plasticity *Physiol. Rev.* **88** 769

[50] Arthur J V and Boahen K 2004 Recurrently connected silicon neurons with active dendrite for one-shot learning *2004 IEEE Int. Joint Conf. on Neural Networks (IEEE Cat. No. 04CH37541)* vol 3

[51] Hidaka M and Akers L A 1991 An artificial neural cell implemented with superconducting circuits *Supercond. Sci. Technol.* **4** 654

[52] Hidaka M and Akers L A 1993 A superconducting neural cell suitable for large scale neural networks *Appl. Supercond.* **1** 1907

[53] J Clarke and A I Braginski (eds) 2006 *The Squid Handbook* (Weinheim: Wiley-VCH)

[54] Schneider M L and Segall K 2020 Fan-out and fan-in properties of superconducting neuromorphic circuits *J. Appl. Phys.* **128** 214903

[55] Litwin-Kumar A, Harris K D, Axel R, Sompolinsky H and Abbott L F 2017 Optimal degrees of synaptic connectivity *Neuron* **93** 1153

[56] Chameh H M, Rich S, Wang L, Chen F-D, Zhang L, Carlen P L, Tripathy S J and Valiante T A 2021 Diversity amongst human cortical pyramidal neurons revealed via their sag currents and frequency preferences *Nat. Commun.* **12** 2497

[57] Braitenberg V and Schuz A 1998 *Cortex: Statistics and Geometry of Neuronal Connectivity* (Berlin: Springer)
[58] Watts D J and Strogatz S H 1998 Collective dynamics of small-world networks *Nature* **393** 440
[59] Sporns O 2010 *Networks of the Brain* 1st edn (Cambridge, MA: MIT Press)
[60] Bassett D S, Greenfield D L, Meyer-Lindenberg A, Weinberger D R, Moore S W and Bullmore E T 2010 Efficient physical embedding of topologically complex information processing networks in brains and computer circuits *PLoS Comput. Biol.* **6** 1
[61] Mahowald M 1992 VLSI analogs of neuronal visual processing: a synthesis of form and function *PhD Dissertation* California Institute of Technology
[62] Boahen K A 2000 Point-to-point connectivity between neuromorphic chips using address events *IEEE Trans. Circuits Syst.* II **47** 416
[63] Park J, Yu T, Joshi S, Maier C and Cauwenberghs G 2017 Hierarchical address event routing for reconfigurable large-scale neuromorphic systems *IEEE Trans. Neural Netw. Learn. Syst.* **28** 2408
[64] Schemmel J, Brüderle D, Grübl A, Hock M, Meier K and Millner S 2010 A wafer-scale neuromorphic hardware system for large-scale neural modeling *2010 IEEE Int. Symp. on Circuits and Systems*
[65] Kumar A, Wan Z, Wilcke W W and Iyer S S 2017 Toward human-scale brain computing using 3D wafer scale integration *ACM J. Emerg. Technol. Comput. Syst.* **13** 45
[66] Cerebras Systems, Inc. 2021 Cerebras (available at: https://cerebras.net/)
[67] Tolpygo S K 2016 Superconductor digital electronics: scalability and energy efficiency issues *Low Temp. Phys.* **42** 361
[68] Miller D A B 1989 Optics for low-energy communication inside digital processors: quantum detectors, sources and modulators as efficient impedance converters *Opt. Lett.* **14** 146
[69] Miller D A B 2000 Rationale and challenges for optical interconnects to electronic chips *Proc. IEEE* **88** 728
[70] Szelag B *et al* 2019 Hybrid III–V/silicon technology for laser integration on a 200-mm fully CMOS-compatible silicon photonics platform *IEEE J. Sel. Top. Quantum Electron.* **25** 8201210
[71] Han Y, Xue Y, Yan Z and Lau K M 2021 Selectively grown III–V lasers for integrated Si-photonics *J. Lightwave Technol.* **39** 940
[72] Gol'tsman G N *et al* 2001 Picosecond superconducting single-photon optical detector *Appl. Phys. Lett.* **79** 705
[73] Marsili F *et al* 2013 Detecting single infrared photons with 93% system efficiency *Nat. Photon.* **7** 210
[74] Buckley S M, Tait A N, Chiles J, McCaughan A N, Khan S, Mirin R P, Nam S W and Shainline J M 2020 Integrated-photonic characterization of single-photon detectors for use in neuromorphic synapses *Phys. Rev. Appl.* **14** 054008
[75] Buckley S, Chiles J, McCaughan A N, Moody G, Silverman K L, Stevens M J, Mirin R P, Nam S W and Shainline J M 2017 All-silicon light-emitting diodes waveguide-integrated with superconducting single-photon detectors *Appl. Phys. Lett.* **111** 141101
[76] Buckley S M *et al* 2020 Optimization of photoluminescence from W centers in silicon-on-insulator *Opt. Express* **28** 16057
[77] Chiles J, Buckley S, Nader N, Nam S W, Mirin R P and Shainline J M 2017 Multi-planar amorphous silicon photonics with compact interplanar couplers, cross talk mitigation and low crossing loss *APL Photonics* **2** 116101
[78] Chiles J, Buckley S M, Nam S W, Mirin R P and Shainline J M 2018 Design, fabrication and metrology of 10×100 multi-planar integrated photonic routing manifolds for neural networks *APL Photonics* **3** 106101
[79] Chiles J *et al* 2018 Deuterated silicon nitride photonic devices for broadband optical frequency comb generation *Opt. Lett.* **43** 1527
[80] Shainline J M 2021 Optoelectronic intelligence *Appl. Phys. Lett.* **118** 160501
[81] McCaughan A N, Verma V B, Buckley S M, Allmaras J P, Kozorezov A G, Tait A N, Nam S W and Shainline J M 2019 A superconducting thermal switch with ultrahigh impedance for interfacing superconductors to semiconductors *Nat. Electron.* **2** 451–6
[82] Schemmel J, Briiderle D, Griibl A, Hock M, Meier K and Millner S 2010 A wafer-scale neuromorphic hardware system for large-scale neural modeling *Proc. 2010 IEEE Int. Symp. Circuits and Systems* pp 1947–50
[83] Benjamin B V *et al* 2014 Neurogrid: a mixed-analog-digital multichip system for large-scale neural simulations *Proc. IEEE* **102** 699–716
[84] Neckar A, Fok S, Benjamin B V, Stewart T C, Oza N N, Voelker A R, Eliasmith C, Manohar R and Boahen K 2018 Braindrop: a mixed-signal neuromorphic architecture with a dynamical systems-based programming model *Proc. IEEE* **107** 144–64
[85] Merolla P A *et al* 2014 A million spiking-neuron integrated circuit with a scalable communication network and interface *Science* **345** 668–73
[86] Davies M *et al* 2018 Loihi: a neuromorphic manycore processor with on-chip learning *IEEE Micro* **38** 82–99
[87] Amit D J 1992 *Modeling Brain Function: The World of Attractor Neural Networks* (Cambridge: Cambridge University Press)
[88] Van Duzer T and Turner C W 1998 *Principles of Superconductive Devices and Circuits* 2nd edn (Englewood Cliffs, NJ: Prentice-Hall)
[89] Kadin A M 1999 *Introduction to Superconducting Circuits* 1st edn (New York: Wiley)
[90] Segall K *et al* 2021 Implementing biological functionality with Josephson junction neurons (in preparation)
[91] Gerstner W and Kistler W 2002 *Spiking Neuron Models* 1st edn (Cambridge: Cambridge University Press)
[92] Koch C and Segev I 2000 The role of single neurons in information processing *Nat. Neurosci.* **3** 1171
[93] Azarfar A, Calcini N, Huang C, Zeldenrust F and Celikel T 2018 Neural coding: a single neurons perspective *Neurosci. Behav. Rev.* **94** 238
[94] Thorpe S, Delorme A and Van Rullen R 2001 Spike-based strategies for rapid processing *Neural Netw.* **14** 715
[95] Johnston D, Magee J C, Colbert C M and Christie B R 1996 Active properties of neuronal dendrites *Annu. Rev. Neurosci.* **19** 165
[96] Salinas E and Sejnowski T J 2001 Correlated neuronal activity and the flow of neural information *Nat. Rev. Neurosci.* **2** 539
[97] Rabinovich M, Volkovskii A, Lecanda P, Huerta R, Abarbanel H D I and Laurent G 2001 Dynamical encoding by networks of competing neuron groups: winnerless competition *Phys. Rev. Lett.* **87** 068102
[98] Yang T and Shadlen M N 2007 Probabilistic reasoning by neurons *Nature* **447** 1075
[99] Ma W J, Beck J M, Latham P E and Pouget A 2006 Bayesian inference with probabilistic population codes *Nat. Neurosci.* **9** 1432
[100] Beck J M, Ma W J, Kiani R, Hanks T, Churchland A K, Roitman J, Shadlen M N, Latham P E and Pouget A 2008 Probabilistic population codes form Bayesian decision making *Neuron* **60** 1142

[101] Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez A N, Kaiser L and Polosukhin I 2017 Attention is all you need (arXiv:1706.03762)
[102] He K, Zhang X, Ren S and Sun J 2016 Deep residual learning for image recognition *Proc. IEEE Conf. on Computer Vision and Pattern Recognition* pp 770–8
[103] De Escobar A L, Hitt R, Mukhanov O and Littlefield W 2006 High performance HF-VHF all digital RF receiver tested at 20 GHz clock frequencies *MILCOM 2006–2006 IEEE Military Communications Conf.* (IEEE) pp 1–6
[104] Kalantre S S, Zwolak J P, Ragole S, Wu X, Zimmerman N M, Stewart M D and Taylor J M 2019 Machine learning techniques for state recognition and auto-tuning in quantum dots *npj Quantum Inf.* **5** 1–10
[105] Prezioso M, Merrikh-Bayat F, Hoskins B D, Adam G C, Likharev K K and Strukov D B 2015 Training and operation of an integrated neuromorphic network based on metal-oxide memristors *Nature* **521** 61–64
[106] Dotson N M and Gray C M 2016 Experimental observation of phase-flip transitions in the brain *Phys. Rev.* E **94** 042420
[107] Chen Y 2017 Mechanisms of winner-take-all and group selection in neuronal spiking networks *Front. Comput. Neurosci.* **11** 20
[108] Lynch N, Musco C and Parter M 2019 Winner-take-all computation in spiking neural networks (arXiv:1904.12591)
[109] McCaughan A N, Oh D M and Nam S W 2019 A stochastic SPICE model for superconducting nanowire single photon detectors and other nanowire devices *IEEE Trans. Appl. Supercond.* **29** 2200104
[110] Izhikevich E M 2004 Which model to use for cortical spiking neurons? *IEEE Trans. Neural Netw.* **15** 1063–70
[111] Young A R, Dean M E, Plank J S and Rose G S 2019 A review of spiking neuromorphic hardware communication systems *IEEE Access* **7** 135606–20
[112] Jaeger H and Haas H 2004 Harnessing nonlinearity: predicting chaotic systems and saving energy in wireless communication *Science* **304** 78–80
[113] Maass W, Natschläger T and Markram H 2002 Real-time computing without stable states: a new framework for neural computation based on perturbations *Neural Comput.* **14** 2531–60
[114] Fourati R, Ammar B, Sanchez-Medina J and Alimi A M 2018 Unsupervised learning in reservoir computing for EEG-based emotion recognition (arXiv:1811.07516 [cs])
[115] Fernando C and Sojakka S 2003 Pattern recognition in a bucket *Advances in Artificial Life* vol 2801 (Berlin: Springer) pp 588–97
[116] Tanaka G, Yamane T, Héroux J B, Nakane R, Kanazawa N, Takeda S, Numata H, Nakano D and Hirose A 2019 Recent advances in physical reservoir computing: a review *Neural Netw.* **115** 100–23
[117] Rowlands G E, Nguyen M-H, Ribeill G J, Wagner A P, Govia L C G, Barbosa W A S, Gauthier D J and Ohki T A 2021 Reservoir computing with superconducting electronics (arXiv:2103.02522 [cond-mat, physics: physics])
[118] Fujimaki A, Nakajima K and Sawada Y 1987 Spatiotemporal observation of the soliton-antisoliton collision in a Josephson transmission line *Phys. Rev. Lett.* **59** 2895–8
[119] Wiesenfeld K, Colet P and Strogatz S 1998 Frequency locking in Josephson arrays: connection with the Kuramoto model *Phys. Rev.* E **57** 1563–9
[120] Nagaoka I, Tanaka M, Inoue K and Fujimaki A 2019 29.3 A 48GHz 5.6mW gate-level-pipelined multiplier using single-flux quantum logic *2019 IEEE Int. Solid-State Conf.—(ISSCC)* pp 460–2
[121] Peng X, Xu Q, Kato T, Yamanashi Y, Yoshikawa N, Fujimaki A, Takagi N, Takagi K and Hidaka M 2015 High-speed demonstration of bit-serial floating-point adders and multipliers using single-flux-quantum circuits *IEEE Trans. Appl. Supercond.* **25** 1301106
[122] Maass W 2011 Liquid state machines: motivation, theory and applications *Computability in Context* (London: Imperial College Press) pp 275–96
[123] Ma Q, Shen L and Cottrell G W 2017 Deep-ESN: a multiple projection-encoding hierarchical reservoir computing framework (arXiv:1711.05255v1)