Harnessing stochasticity for superconductive multi-layer spike-rate-coded neuromorphic networks

NEUROMORPHIC
Computing and Engineering

PAPER

Harnessing stochasticity for superconductive multi-layer spike-rate-coded neuromorphic networks

Alexander J Edwards[1,3,*], **Gleb Krylov**[2,*], **Joseph S Friedman**[1] and **Eby G Friedman**[2]

[1] The University of Texas at Dallas, Richardson, TX, United States of America
[2] University of Rochester, Rochester, NY, United States of America
[3] First authors contributed equally to this work.
[*] Authors to whom any correspondence should be addressed.

E-mail: Alexander.Edwards@utdallas.edu and Gleb.Krylov@tum.de

Keywords: neuromorphic computing, stochasticity, single flux quantum logic, superconductor electronics

OPEN ACCESS

RECEIVED
4 August 2023

REVISED
9 November 2023

ACCEPTED FOR PUBLICATION
19 January 2024

PUBLISHED
23 February 2024

Abstract

Conventional semiconductor-based integrated circuits are gradually approaching fundamental scaling limits. Many prospective solutions have recently emerged to supplement or replace both the technology on which basic devices are built and the architecture of data processing. Neuromorphic circuits are a promising approach to computing where techniques used by the brain to achieve high efficiency are exploited. Many existing neuromorphic circuits rely on unconventional and useful properties of novel technologies to better mimic the operation of the brain. One such technology is single flux quantum (SFQ) logic—a cryogenic superconductive technology in which the data are represented by quanta of magnetic flux (fluxons) produced and processed by Josephson junctions embedded within inductive loops. The movement of a fluxon within a circuit produces a quantized voltage pulse (SFQ pulse), resembling a neuronal spiking event. These circuits routinely operate at clock frequencies of tens to hundreds of gigahertz, making SFQ a natural technology for processing high frequency pulse trains. This work harnesses thermal stochasticity in superconducting synapses to emulate stochasticity in biological synapses in which the synapse probabilistically propagates or blocks incoming spikes. The authors also present neuronal, fan-in, and fan-out circuitry inspired by the literature that seamlessly cascade with the synapses for deep neural network construction. Synapse weights and neuron biases are set with bias current, and the authors propose multiple mechanisms for training the network and storing weights. The network primitives are successfully demonstrated in simulation in the context of a rate-coded multi-layer XOR neural network which achieves a wide classification margin. The proposed methodology is based solely on existing SFQ technology and does not employ unconventional superconductive devices or semiconductor transistors, making this proposed system an effective approach for scalable cryogenic neuromorphic computing.

1. Introduction

Conventional level-based artificial neural networks (ANNs)—while useful in a large number of applications—suffer from high computational costs that may be mitigated by using alternative biomimetic architectures such as spiking neuromorphic networks (SNNs) [1]. As the human brain can seemingly compute similar computation at a fraction of the energy, recent attention has gravitated towards biomimetic hardware, emulating biology in both phenomenological behavior and emergent computation. Mimicking biological neuronal spiking behavior, SNNs are a class of neural networks in which data are encoded in a sequence of temporal spikes as opposed to a single real-valued signal (illustrated in figure 1); SNNs are therefore highly attractive for low power neuromorphic hardware.

Operating on minimal voltage pulses that function like neuronal spiking events, single flux quantum (SFQ) systems are particularly attractive for developing biomimetic SNNs. Superconducting Josephson

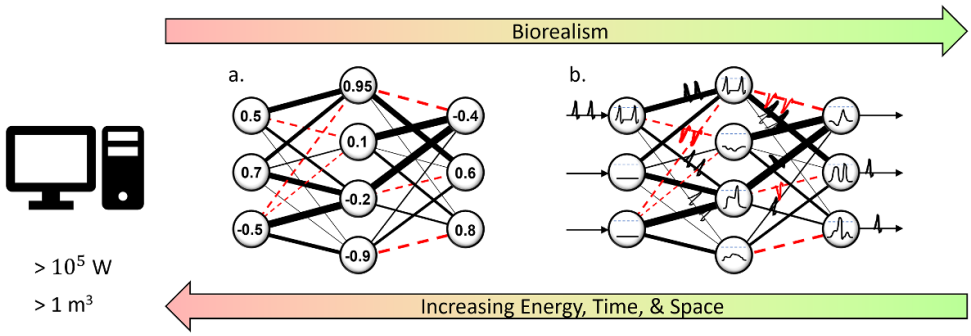


Figure 1. Biological inspiration for SNNs with SFQ. (a) Conventional artificial neural network. Information propagates through the network from left to right. The output states of the neurons (circles) are encoded by a single real-valued number which is modulated by downstream synapses (black and red lines) which are collected by the successive (post-synaptic) neurons according to an activation function. (b) Spiking neural network (SNN). Rather than representing information as a single number, spiking neurons mimic biology by retransmitting spikes downstream, which are modulated by the synapse strength. The information representation can therefore be much richer, enabling systems approaching the incredible efficiency observed in biology.

junctions (JJs) inherently react to and regenerate fluxons, enabling extremely energy efficient neuronal and synaptic circuits. Whereas modern large scale SNNs utilize inefficient packet-based routing networks [2–4], SFQ may be ideally suited for ultra-fast, ultra-low power SNN systems. The similarities between SFQ pulses and neuronal spiking have been explored in the literature [5–17], with synapses modifying the amplitude of spiking events.

In this work, we take a novel approach toward SFQ neuromorphic networks by using a stochastic gate as a synapse along with spike-rate-coding [18–20] to encode information in the collective spike rate instead of precise spike timing. This system is therefore more resilient to noise but may incur longer delays as the outputs need more time to stabilize. We propose neuronal and synaptic SFQ network primitives that may be cascaded with minimal layer-to-layer circuit design to construct spike-rate-coded networks in various neural network architectures.

SFQ synapses emulating the stochasticity in the brain [19, 21–23] are described along with SFQ leaky-integrate-and-fire (LIF) neurons. These primitive synaptic and neuronal circuits directly cascade with each other, permitting the construction of deep neuromorphic networks. The primitive circuits may be likewise amenable to various data encodings and learning mechanisms, although this work is primarily concerned with using these primitive circuits in a supervised, offline-learning context with spike-rate-coding. A multilayer network computing the XOR functionality is constructed and simulated showing high non-linearity and input separation. Incidentally, this is the first multi-layer fully-SFQ SNN demonstration in the literature, although larger single-layer spiking SFQ networks have been demonstrated both in simulation [13, 18] and experimentally [16, 17].

2. Background

SNNs and SFQ circuits have received significant attention in the literature including proposals for SNN primitives implemented with SFQ, as we summarize in the following subsections.

2.1. Spiking neuromorphic networks

SNNs attempt to replicate biological spiking behavior so as to unlock the ultra-low-power computation observed in biological systems. Biological neural networks encode information based on the frequency and relative timing of neuronal spiking events. This behavior is counter to conventional software neural networks, which represent information as single-valued numbers. Spiking neuromorphic networks are an attempt to more richly encode information with equivalent or cheaper hardware for more efficient, biomimetic computation.

Neurons in SNNs mimic biology by aggregating spiking activity from upstream neurons and firing when sufficiently stimulated, creating a spike event that is propagated downstream. A common neuronal model, the LIF neuron, integrates input spikes into an internal potential which leaks over time if not stimulated. When the potential crosses the action potential threshold, the neuron fires, resets, and begins integrating again. The LIF model is a very common neuronal model for SNNs [1].

2.2. Stochastic synapses

Synapses in SNNs mimic biology by modulating the strengths of the connections between upstream and downstream neurons. While most SNNs employ synapses that modulate the *amplitude* of spiking events, this work utilizes an alternative approach wherein synapses modulate spike *rate* by—inspired by the prevalent stochasticity observed in biological systems [19, 21–23] – *stochastically* gating individual synaptic spiking events.

Stochasticity in synapses has appeared throughout the literature for both spiking [19, 20, 22, 24] and level-based networks [25, 26]. In level-based networks synapse weights are sampled several times per inference in a neural sampling machine whereas in spiking networks, stochastic synapses are effectively integrated with spike-rate information encodings [19, 20]. In this work, synapses stochastically pass a fraction of incoming spikes with a probability w, known as 'blank-out' gating [19, 20, 25]. Given an input spike train following a Poisson process with rate λ_{in}, the synapse output spike rate is given by: $\lambda_{\text{out}} = w * \lambda_{\text{in}}$, similarly following a Poisson process [19, 20].

2.3. Spike rate coding

Spike-rate-coded networks are a subset of SNNs in which information is encoded solely in average neuronal spike rates on wires throughout the network. Spike rate coded spiking neural networks may be trained similar to level-based feed-forward ANNs via backpropagation, assuming the following:

- information is only coded in the average spike rate and no useful information is present in the spike timing,
- the synapses modulate the spike rate by a trainable constant multiplier, and
- the neuronal output spike rate depends only on the input spike rate according to a continuous transfer or activation function [19].

Under these assumptions, the output rate of a synapse randomly passing a fixed proportion of fluxons is equal to multiplying the input rate by that synapse weight [19, 20]. Furthermore, confluence of post-synaptic spike trains result in a single spike train whose rate is the sum of the input rates. This rate may saturate at large frequencies which may be treated like saturation of an activation function. There therefore exists a one-to-one mapping between feed-forward spike-rate-coded SNNs and level-based ANNs, enabling identical training via backpropagation for both networks.

The benefits of this equivalence are manifold: in the case where spiking SFQ networks may be lower cost than level-based ANNs, previously trained networks may be based on SFQ hardware, and new networks may be effectively trained with extant software tools. Furthermore, backpropagation under spike-rate-coding assumptions may be used to train starting point networks for advanced learning techniques that may be evolved using unsupervised Hebbian learning techniques to begin encoding information in spike timing, further approaching the richness of information coding in the brain.

2.4. Single flux quanta circuitry

Single flux quantum logic is an emerging cryogenic technology for highly energy efficient computing [27]. SFQ circuits are based on JJs and superconducting quantum interference devices (SQUIDs), and operate with magnetic flux quanta. The quanta are typically represented by voltage pulses of quantized area equal to the magnetic flux quantum ($\Phi_0 \sim 2.07\,\text{mV} \cdot \text{ps}$) [28]. These pulses are generated in a process often referred to as JJ switching—a shift of superconducting phase between the terminals of the JJ by 2π.

The primary advantage of SFQ circuits for digital logic is the unparalleled energy efficiency and speed. Each 2π transition of a typical 100 μA JJ dissipates energy on the order of 2×10^{-19} J [29]. Although a logic operation requires several switches, the energy per operation is several orders of magnitude lower than state-of-the-art CMOS logic even considering the cryogenic cooling to 4.2 K (liquid helium temperature) [30]. Additionally, as SFQ circuits operate on quantized pulses instead of voltage levels, SFQ systems are essentially very deep pipelines, as many distinct pulses can concurrently travel along different points of the same wire. This deep pipelining enables spike frequencies from tens to hundreds of gigahertz, one to two orders of magnitude higher than the fastest CMOS ASICs.

Whereas in conventional SFQ logic, information is encoded as the presence or absence of an SFQ pulse within a specific time period, alternative JJ-based circuits can generate and operate on more complex SFQ pulse sequences. This includes the generation of pulse sequences with a controllable frequency and stochastic switching induced by thermal noise. Both properties are exploited in this work.

Several approaches for neuromorphic computing with superconducting circuits have been proposed, although none utilize stochasticity in the synapses. A non-spiking superconducting XOR neural network was experimentally demonstrated [31]. Several proposals demonstrate individual SFQ gates for neuronal [5–7, 15], synaptic [7–9, 32], or interconnect [10] functionality. Two-neuron oscillatory Hopfield networks [11,

12] and single layer feed-forward spiking networks [13, 16–18] have been proposed, and a multi-layer feed-forward spiking network with heterogeneous CMOS-SFQ circuitry has been demonstrated in simulation [14]. Some of the proposed approaches utilize magnetic JJs to gradually modify the internal state of the gates, enabling online learning [8, 13, 16]. Fabrication processes used to manufacture these devices are, however, not well established, and the resulting circuits are limited in scale. In [33], an SFQ-based methodology for building neuromorphic networks is proposed, where a bipolar current is used to represent a logic state, which is converted into a train of SFQ pulses for transmission. Stochasticity in neuromorphic SFQ neurons has been studied [17], but to the authors' best knowledge, there are no studies exploring stochasticity in SFQ synapses.

3. Stochastic superconducting neuromorphic primitive circuits for spike-rate-coded networks

We propose cascadable primitive SFQ synaptic and neuronal circuits for building large multi-layer neuromorphic networks. The synapse is stochastic, based on a Josephson balanced comparator [34], and the pulse trains are briefly converted into magnetic flux within neurons before undergoing a threshold to produce an output pulse train. The spiking SFQ neurons integrate incoming spikes and fire when sufficiently stimulated, and as all information about the network state is encoded solely in the neuronal and synaptic spiking activity, the resultant network is an SNN. The network uses established superconductive fabrication processes [35] and is tuned by an external current, facilitating large scale integration. All of the circuits presented here are simulated in WRspice [36] based on the state-of-the-art MIT LL SFQ5ee fabrication process [35].

3.1. Data encoding with SFQ

Because of the phenomenological similarity between neuronal spikes and SFQ pulses, information in the network is represented in the timing and frequencies of SFQ pulses. Upon sufficiently stimulating a JJ with voltage and/or bias current, a 2π phase shift is created around a superconducting loop, producing a corresponding SFQ voltage pulse across the JJ, which can be propagated to stimulate downstream JJs. As an SFQ is the smallest possible non-zero voltage pulse, fluxons are ideal information carriers for SNNs, enabling extremely low power neuromorphic processing. Furthermore, information in a sequence of fluxons is not encoded in the magnitude of the spike but rather in the spike timing as is the case in SNNs and, more notably, the brain. Fluxons have non-volatile attributes as well, enabling short-term memory and the construction of low power LIF neurons.

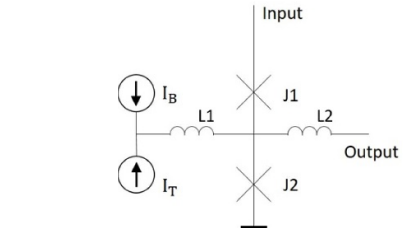
A wide range of SNN topologies and encodings is available, and this work is particularly focused on spike-rate-coded architectures because of simplified training (section 2.3) and robustness to the stochasticity of the synapses (section 3.2) although the hardware here is amenable to other encoding methodologies. Among the various SNN topologies are perceptron networks and feed-forward deep neural networks; both of which are amenable to spike-rate-coding for classification tasks. Recurrent neural networks may be constructed as well, enabled by internal neuronal time delays in fluxon propagation and encoding information in timing between spikes. Strong biological evidence suggests that stochastic synapses may be well-suited for these timing-coded networks [19, 21–23] although more information is lost by not propagating an input spike, indicating that online learning using spike timing—such as spike-time-dependent plasticity [1]—is also amenable to SFQ networks [32] and may similarly ease training costs.

3.2. Stochastic-pass synapses

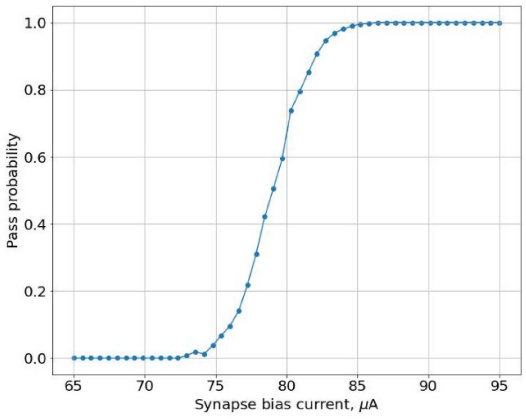
Inspired by the stochasticity in biological systems [19, 21–23], the proposed synapse stochastically gates incoming spikes, encoding the synaptic weight in the probability that the synapse will propagate an incoming fluxon to the output, implementing the first SFQ 'blank-out' [19, 20, 25] synapse in the literature. Stochastically dropping spikes at synapses is a well-known phenomenon in biology [19, 21–23], and biomimetic hardware will likely lead to efficient hardware.

These stochastic synapses are amenable to rate-coded networks as they directly modulate the rate of a spike-train as described in section 2.2. Additionally, as stochastic biological systems often utilize spike-timing to encode to information, it is conceivable that future neuromorphic *timing-coded* SNNs would benefit from stochasticity like that proposed here, although the exact function of such stochasticity in biological systems has yet to be understood completely and is not explored further here.

Whereas other synapse approaches modulate the effective amplitude of spiking events, due to the quantized nature of fluxons, it is difficult to accomplish this modulation without conversion between the fluxons and analog current, thereby decreasing system efficiency when amplitude-weighting is used. In the



(a)



(b)

Figure 2. Synaptic circuit, (a) schematic, and (b) graph of pass-probability with bias current. The data are presented for $I_C(J_1, J_2) = 150\ \mu A$, input SFQ pulse frequency of 50 GHz, and T = 4.2 K.

stochastic scheme, as both the inputs and outputs to synapses are fluxons, these conversions are not necessary making them potentially more efficient in scaled systems. The corresponding trade-off is that incorporating stochasticity into network behavior requires an increase in redundancy, whether longer spike-trains in rate-coded networks or redundant data-paths in timing-coded networks.

The proposed synapse is constructed from a Josephson balanced comparator [34, 37], harnessing thermal noise for true stochasticity. A Josephson balanced comparator is a pair of serially connected JJs with the bias current, I_B, applied between these JJs, as shown in figure 2(a). When an SFQ pulse is applied to the input, one of the JJs within the comparator undergoes a 2π phase shift, depending on the magnitude of I_B. For small I_B, $J1$ switches, absorbing the input pulse, whereas for large I_B, $J2$ switches, propagating the input pulse to the output. In the absence of noise current I_T, the gating functionality depends deterministically on the applied currents, however in the presence of thermal fluctuations in the applied currents, the balanced comparator exhibits stochastic behavior—a 'grey zone' [34]. A similar approach has been proposed for SFQ based synapses [33], where a C-SQUID [38] is used rather than a balanced comparator.

The synapse weight is encoded in the probability that an input spike will propagate to the output and can be modulated after fabrication by adjusting I_B. Figure 2(b) depicts the probability of passing an incoming SFQ pulse to the output as a function of bias current for specific circuit parameters, demonstrating a continuous swing of weights between 0 (all fluxons blocked) and 1 (all fluxons propagated). The range of synapse weights may be extended to $[-1, 1]$ using differential pairs of synapses connected to the excitatory and inhibitory inputs of each neuron (see section 3.3).

3.3. Leaky integrate-and-fire neurons

Neuronal LIF circuits—integrating incoming SFQ pulses until an internal threshold is reached and outputting a resultant SFQ spike sequence—are described here. A leaky SQUID loop is inductively coupled with a firing SQUID loop [39] to implement integrating, leaking, and firing behavior.

A conventional SQUID loop with a large inductance is used to integrate incoming SFQ pulses. The input SFQ pulses are applied across $J1$ or $J2$ (indicated in figure 3) as excitatory or inhibitory inputs respectively. As $J1$ switches, the fluxon energy is stored as current in inductor $L1$. This energy is accumulated with additional inputs increasing the flux stored in the loop. A fluxon applied to the inhibitory input switches $J2$, reducing the flux stored in the loop and allowing neurons to have both excitatory and inhibitory interactions.

Leaking may be implemented with the addition of a resistive element R dissipating the energy stored in inductor $L1$. A resistance on the order of a few ohms produces a linear leakage characteristic, the timing of

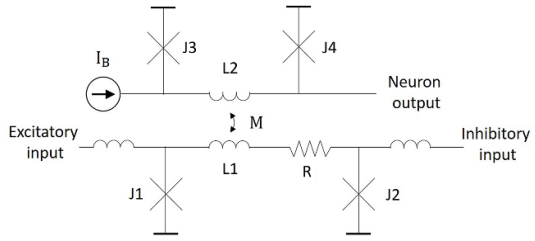


Figure 3. Neuronal circuit.

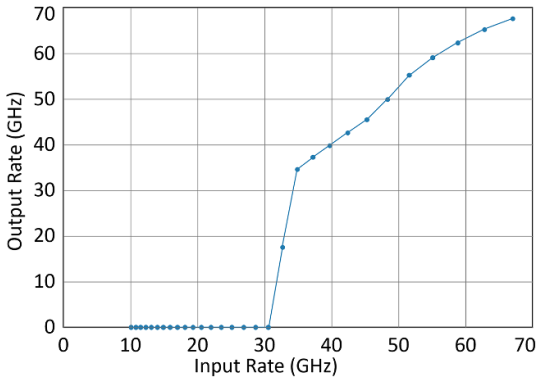


Figure 4. Nonlinear neuron activation function, showing output rate as a function of input rate. The neuron begins activating near an input spike rate of 33 GHz.

which can be tuned by changing the size of the resistor. Utilizing the dynamic SFQ (DSFQ) leakage mechanism, additional JJs may be introduced into the loop to provide a faster reset of the circulating current [40, 41].

When the current in the SQUID is sufficiently large, an inductively coupled SQUID produces a firing pulse. A JJ partially biased by an inductively coupled neuron loop current, is shown in figure 3. Due to the coupling between $L1$ and $L2$, as $L1$ integrates input pulses, current in $L2$ simultaneously increases. Eventually current in $L2$—with contributions from $L1$ and I_B—is sufficiently large to switch $J4$ producing an output pulse. In the case where inhibitory inputs are more frequent than excitatory inputs, $J3$ will switch instead of $J4$ sans output spike. Bias currents through the JJs help to regenerate fluxons, enable signal fan-out, and improve the cascade characteristics.

The relationship between the neuron input and output rates is non-linear, a necessary condition for proper neural network functionality. Figure 4 depicts the relationship between input and output spike rates (equivalent to the activation function of a conventional level-based neuron if spike-rate-coding is employed). Note the distinct non-linearity akin to a rectified linear unit (RELU), scaled exponential linear unit (SELU), or sigmoidal activation function. The width and slope of the threshold region as well as the saturation characteristics are adjusted by tuning I_B, $L1$, $L2$, R, and $I_C(J1)$ (see figure 3). The threshold input rate may be tuned after fabrication through the application of bias input sequences. At higher frequencies, output rate saturates as $J4$ moves toward the resistive regime adding additional non-linearity.

The input fan-in is realized using conventional RSFQ confluence buffers (pulse mergers) [28]. These buffers exhibit a saturating spike rate, as multiple input pulses arriving in close succession produce only one output pulse. This property assists in the saturation of the neuronal output spike rate.

Negative weights are applied as inhibitory inputs to the neuron. A single synapse can therefore be implemented as a differential pair of synapse circuits connected to the excitatory and inhibitory inputs of a neuron. Inhibitory or excitatory bias input spike sequences may be added to adjust the neuron threshold.

Fan-out of signals is challenging, although solutions for SFQ fan-out have been proposed in the literature. As a fluxon is minimal and cannot be split, fluxons must be regenerated whenever a transmission line branches. Fan-out is therefore managed by splitter trees, which could incur well-known area and delay overhead [10, 18]. Multiple techniques exist to reduce the overhead of the signal fan-out in large scale SFQ circuits [42, 43]. These techniques are primarily based on utilizing a splitter with more than two outputs at the cost of reduced parameter margins. Large-scale fan-out trees are therefore possible although costly, and

sparse network architectures should be employed. This constraint therefore encourages the deployment of pre-trained networks that may be thoroughly pruned before SFQ design as described in section 3.5.

3.4. Cascading synapses and neurons for deep networks

The SFQ neuromorphic primitives can be cascaded to construct deep neuromorphic networks enabling straightforward circuit design with minimal layer-to-layer tuning. As inputs and outputs from synapses, neurons, fan-in, and fan-out circuits are all SFQ pulses there is no need for costly signal conversion to construct large multilayer networks, and there is no need for a clock network as all of the primitive circuits operate asynchronously.

In terms of reliability, the stochasticity of the synapses will not cause unwanted noise in downstream neuronal, fan-in, and fan-out circuits. The synapses are biased to a stochastic grey zone such that thermal energy at 4.2 K is sufficient for synaptic stochasticity, whereas the neuronal, fan-in, and fan-out circuits are designed to be reliable at these standard SFQ temperatures. As synapses do not create any additional noise, the only thermal noise that the synapse propagates to the rest of the circuit is via the spike rate and timing, which is the intended function of the synapses.

To ensure that spiking activity is similar from layer to layer, spiking activity can be regenerated through the use of signal confluence and additional input spike sequences, mitigating that the synapse output spike rate will always be less than or equal to the input rate. When an RSFQ confluence buffer merges two or more spike sequences, the output spike rate is the combined rate of the input sequences, increasing signal activity. Furthermore, additional bias spike sequences may be incorporated to further regenerate signal activity.

Layer-to-layer spiking activity can be tuned at a network-architecture level, mitigating the need to individually tune circuit parameters for each layer or for a specific network architecture. Specifically, synapse weights should be tuned to ensure that spiking activity remains within proper regions of operation for the circuits. Additionally, the rates of the bias input spike sequences may be tuned through the use of synapse circuits, and neuronal bias currents may be programmed to adjust the neuronal firing threshold.

3.5. Tuning network weights and biases

The authors propose two ways to set weights and neuronal biases, both of which are controlled by DC current. Firstly, weights may simply be hardcoded during fabrication using standard SFQ bias current distribution techniques such as resistive trees or current-limiting JJs [44]. While the network would not be tunable after fabrication, it can still be useful in mass-produced edge-sensing devices where extreme resource efficiency is crucial, as pre-trained networks can be pruned and optimized before fabrication and do not need tunability, which can incur large energy, area, and routing costs. The case study of section 4 assumes hardcoded weights and biases.

The second possible weight-storage technique involves the circuit shown in figure 5(a), in which a SQUID loop—very similar to that presented in the neuron of figure 3—is inductively coupled to the synapse output inductor, similarly to a blocking gate [44]. As the SQUID loop of figure 5(a) does not incorporate a resistor, the device is non-volatile; fluxons may be added to the SQUID loop through input A or removed by pulsing input B enabling tunable control of the synapse weight. The relative inductance M will control the impact of each stored SFQ pulse on the synapse weight, enabling fine- or coarse-grained precision, and combinations of fine- and coarse-grained circuits can be used to jointly adjust the same synapse for increased weight range and precision. As demonstrated in the simulation of figure 5(b), a tuning circuit connected to the output inductor of a synapse allows for tunable spike pass probability. Finally, as inputs A and B may be connected to other SFQ circuits in the network, this circuit is amenable to direct implementations of online learning in which SFQ pulses may be sent to input A (B) to (depress) potentiate the weight *during network operation.*

4. Case Study: demonstration of two-layer XOR neural network

Superconducting neuromorphic networks with stochastic synapses are demonstrated here with a multi-layer spike-rate-coded architecture. This is incidentally the first demonstration of a multi-layer fully SFQ neuromorphic network, although larger single-layer networks have been explored extensively in simulation and experiment [13, 16–18]. A spike-rate encoding is chosen for ease of training, although the SNN primitives outlined in section 3 are amenable to a wide range of data encodings and network architectures. The network is trained to compute the XOR functionality, and demonstrates large non-linearity and input separation.

4.1. Training and weight mapping of XOR network

A trained neuromorphic network to compute the two bit XOR function requires at least two layers in an ANN along with negative weights and bias inputs. The chosen network is depicted in figure 6. Input neurons

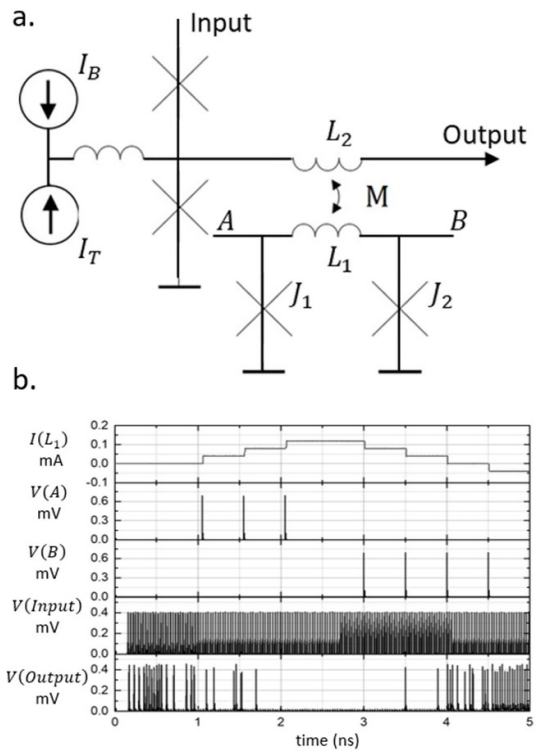


Figure 5. Tunable synapse. (a) The inductive coupling from the SQUID loop (A-J_1-L_1-J_2-B) limits the spikes that can propagate through the synapse. Mutual inductance M specifies the precision and range of the weight. (b) Simulation of the tunable synapse circuit. When the current through inductor L_1 is low (high), spike events are propagated (blocked) by the synapse. Non-volatile storage of fluxons in the loop enable non-volatile setting of weights using inputs A and B that could be amenable to both offline and online learning, post-fabrication.

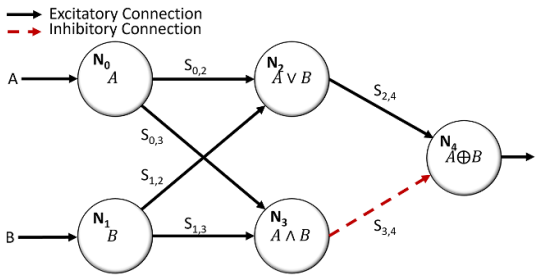


Figure 6. XOR spiking neural network. The primary inputs and primary outputs are represented as spike frequencies in proportion to the saturation output neuronal rate.

0 and 1 normalize the network inputs rates to the neuronal output levels. Due to trained neuronal biases, neuron 2 will have a high output when one of the two inputs is on while neuron 3 requires both inputs to be on in order to activate. Neuron 4 uses inhibitory weights to compute the XOR functionality, activating only if neuron 2 is active while neuron 3 is not. Neuron 4 will therefore only turn on when exactly one of the inputs is on: the XOR functionality. A similar non-spiking architecture was demonstrated experimentally in superconducting hardware in [31].

For this demonstration, spike-rate-coding is chosen to map the SFQ neuromorphic primitives to the chosen network. As described in sections 2.3 and 3.1, spike-rate-coding supports training similar to a level-based ANN and allows direct mapping of architecture and trained weights between the level-based ANN and the spike-rate-coded SNN. Weights are trained offline and are treated as constants in the network simulations. The two network inputs are stationary pulse trains with rates close to neuronal saturating rates, encoding a logic 0 (1) as a low (high) spike rate. The synapse weights are encoded in the probability of the spike propagation. The bias input sequences are trained to make the neurons sensitive to different combinations of inputs, tuning the threshold input rate of figure 4 to ensure large separability between the input patterns. The network outputs may be interpreted by the average spike rate relative to the thresholding spike rate of the neurons, which is close to 33.3 GHz in this circuit.

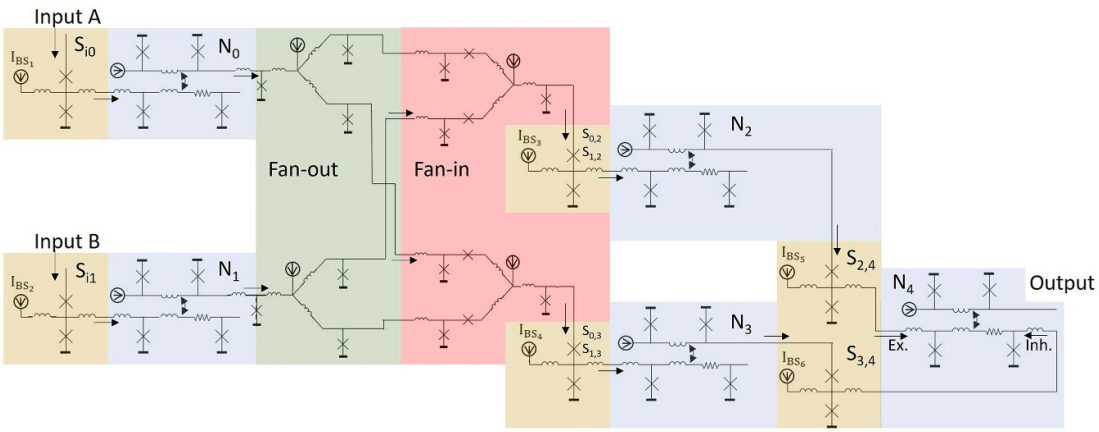


Figure 7. Neuromorphic SFQ network computing a two bit XOR function with synapses in yellow, neurons in blue, fan-out in green, and fan-in in red.

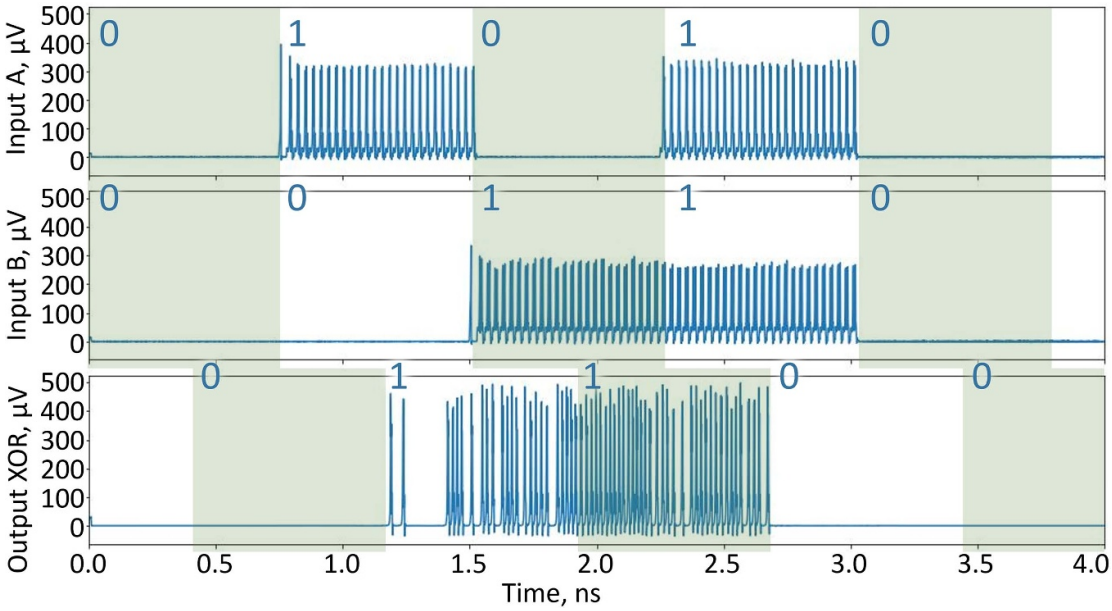


Figure 8. Spiking behavior with different input stimuli for XOR network.

4.2. Circuit topology of multilayer network

The XOR network is directly implemented with the SFQ primitives described in section 3. A schematic of the neuromorphic network based on these components, computing a two bit XOR function (presented in figure 6) is shown in figure 7. The synaptic weights and neuronal biases are tuned by changing the bias current.

4.3. Network results

As shown in figure 8, the network correctly computes the XOR function. Note that the output signals can spike more frequently than the inputs, indicating that although the neurons and synapses reduce the spike rate to downstream devices, the splitters and confluence buffers regenerate spiking activity (see section 3.4), enabling deeper neuromorphic networks. Note also that there is a non-negligible propagation delay between the arrival of the input spikes and the computation of the output; the driving circuitry of the neuromorphic network would therefore need to account for delays from network signal propagation and stabilizing output. As delays and output signal encoding are strongly application- and network architecture-dependent, design of these external circuits is beyond the scope of this work, which is intended to be device, architecture, and encoding agnostic.

Additionally the network is robust to variations in the input rates, showing a large separation between classes. A phase diagram of classification for the XOR network is shown in figure 9(a), where the dependence

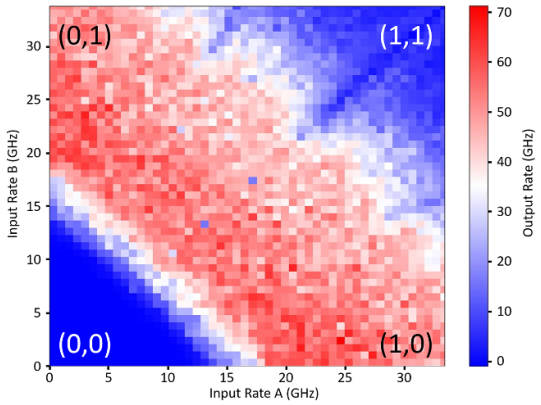


Figure 9. Phase diagram of classification for XOR network output. Red represents output rates substantially greater 33.3 GHz which is the neuronal threshold, blue represents output rates substantially less than 33.3 GHz with the colors in between representing the domain boundaries.

of the network output rate is shown as a function of the two input rates. It is desirable for the high output rates corresponding to the output of logic 1 (shown in figure 9 in red) to map to the input rates corresponding to logic 10 and 01. The input rates corresponding to logic 00 and 11 should produce low spike activity (shown in blue). This classification diagram displays good separation of the output states with respect to the input spike rates. The proposed network is therefore robust to variations in the input rates.

This small network therefore works properly, and it is expected that the same network primitives may be used to design larger networks as described in sections 3.3 and 3.4. Fan-out may be designed using splitter trees as described in [10, 18], and directly implementing pruned, sparse, *pre-trained* networks can reduce the burden of fan-out and save area and energy cost associated with implementing tunable weights (see section 3.5).

5. Conclusions

We propose stochastic synapses for deep superconducting neuromorphic networks. Synapses stochastically gate the propagation of spike events and are especially suitable for rate-coded architectures, although biology and the literature suggest such stochasticity will be similarly well-suited for spike-timing based encodings. SFQ neuromorphic primitive circuits are presented that can be directly cascaded to construct a broad range of network architectures. While the demonstration shows one feed-forward network, the circuits are amenable to richer spiking network architectures and encodings including recurrent neural networks, spike-time-dependent networks, and online learning. As network layers can regenerate spiking behavior, deep network architectures are readily attainable through natural cascading of successive layers.

The proposed network—entirely based in available SFQ technologies—has the advantage of being extremely energy efficient as compared with conventional CMOS technologies [14, 30]. Furthermore, as conversions between fluxons and analog currents are constrained within each individual neuron, the network is compact and scalable. Additionally, the proposed scheme does not require unconventional devices or complex 2.5D or 3D integration, and can be produced using standard niobium fabrication processes.

Data availability statement

The data cannot be made publicly available upon publication because they are not available in a format that is sufficiently accessible or reusable by other researchers. The data that support the findings of this study are available upon reasonable request from the authors.

ORCID iDs

Alexander J Edwards https://orcid.org/0000-0002-7050-3151
Gleb Krylov https://orcid.org/0000-0002-3022-0368
Joseph S Friedman https://orcid.org/0000-0001-9847-4455
Eby G Friedman https://orcid.org/0000-0002-5549-7160

References

[1] Schuman C D, Kulkarni S R, Parsa M, Mitchell J P, Date P and Kay B 2022 Opportunities for neuromorphic computing algorithms and applications *Nat. Comput. Sci.* **2** 10–19
[2] Furber S B, Lester D R, Plana L A, Garside J D, Painkras E, Temple S and Brown A D 2013 Overview of the spinnaker system architecture *IEEE Trans. Comput.* **62** 2454–67
[3] Akopyan F *et al* 2015 Truenorth: design and tool flow of a 65 mw 1 million neuron programmable neurosynaptic chip *IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst.* **34** 1537–57
[4] Davies M *et al* 2018 Loihi: a neuromorphic manycore processor with on-chip learning *IEEE Micro* **38** 82–99
[5] Goteti U S and Dynes R C 2021 Superconducting neural networks with disordered Josephson junction array synaptic networks and leaky integrate-and-fire loop neurons *J. Appl. Phys.* **129** 073901
[6] Yamanashi Y, Umeda K and Yoshikawa N 2013 Pseudo sigmoid function generator for a superconductive neural network *IEEE Trans. Appl. Supercond.* **23** 1701004
[7] Crotty P, Schult D and Segall K 2010 Josephson junction simulation of neurons *Phys. Rev.* E **82** 011914
[8] Schneider M L, Donnelly C A, Haygood I W, Wynn A, Russek S E, Castellanos-Beltran M A, Dresselhaus P D, Hopkins P F, Pufall M R and Rippard W H 2020 Synaptic weighting in single flux quantum neuromorphic computing *Sci. Rep.* **10** 934
[9] 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
[10] Schneider M L and Segall K 2020 Fan-out and fan-in properties of superconducting neuromorphic circuits *J. Appl. Phys.* **128** 214903
[11] 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
[12] 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
[13] Schneider M L, Donnelly C A and Russek S E 2018 Tutorial: high-speed low-power neuromorphic systems based on magnetic Josephson junctions *J. Appl. Phys.* **124** 161102
[14] Bozbey A, Karamuftuoglu M A, Razmkhah S and Ozbayoglu M 2020 Single flux quantum based ultrahigh speed spiking neuromorphic processor architecture (arXiv:1812.10354 [cs.ET])
[15] Karamuftuoglu M A, Bozbey A and Razmkhah S 2023 JJ-Soma: towards a spiking neuromorphic processor architecture *IEEE Trans. Appl. Supercond.* 33 1–7
[16] Jué E, Pufall M R, Haygood I W, Rippard W H and Schneider M L 2022 Perspectives on nanoclustered magnetic Josephson junctions as artificial synapses *Appl. Phys. Lett.* **121** 240501
[17] 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
[18] Schneider M, Toomey E, Rowlands G, Shainline J, Tschirhart P and Segall K 2022 Supermind: a survey of the potential of superconducting electronics for neuromorphic computing *Supercond. Sci. Technol.* **35** 053001
[19] Goldberg D H, Cauwenberghs G and Andreou A G 2001 Probabilistic synaptic weighting in a reconfigurable network of vlsi integrate-and-fire neurons *Neural Netw.* **14** 781–93
[20] Neftci E O, Pedroni B U, Joshi S, Al-Shedivat M and Cauwenberghs G 2016 Stochastic synapses enable efficient brain-inspired learning machines *Front. Neurosci.* **10** 241
[21] Maass W 2015 To spike or not to spike: that is the question *Proc. IEEE* **103** 2219–24
[22] Maass W and Zador A 1997 Dynamic stochastic synapses as computational units *Advances in Neural Information Processing Systems* vol 10, ed M Jordan, M Kearns and S Solla (MIT Press)
[23] Braun H A 2021 Stochasticity versus determinacy in neurobiology: from ion channels to the question of the free will *Front. Syst. Neurosci.* **15** 629436
[24] Wang C, Wang K, Wen X, Luo W, Liang S, Zhang Y and He Y 2022 Stochastic synapses made of magnetic domain walls *Phys. Rev. Appl.* **18** 064014
[25] Dutta S, Detorakis G, Khanna A, Grisafe B, Neftci E and Datta S 2022 Neural sampling machine with stochastic synapse allows brain-like learning and inference *Nat. Commun.* **13** 2571
[26] Shah S N H and Hougen D F 2017 Stochastic synapse reinforcement learning (SSRL) *2017 IEEE Symp. Series on Computational Intelligence (SSCI)*
[27] Krylov G and Friedman E G 2022 *Single Flux Quantum Integrated Circuit Design* (Springer)
[28] 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
[29] Mukhanov O A 2011 Energy-efficient single flux quantum technology *IEEE Trans. Appl. Supercond.* **21** 760–9
[30] Holmes D S, Ripple A L and Manheimer M A 2013 Energy-efficient superconducting computing - power budgets and requirements *IEEE Trans. Appl. Supercond.* **23** 1701610
[31] 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
[32] Segall K, Purmessur C, D'Addario A and Schult D 2023 A superconducting synapse exhibiting spike-timing dependent plasticity *Appl. Phys. Lett.* **122** 242601
[33] Semenov V K, Golden E B and Tolpygo S K 2022 A new family of biosfq logic and memory cells *IEEE Trans. Appl. Supercond.* **32** 1–5
[34] Filippov T and Kornev V 1991 Sensitivity of the balanced Josephson-junction comparator *IEEE Trans. Magn.* **27** 2452–5
[35] Tolpygo S K, Bolkhovsky V, Rastogi R, Zarr S, Day A L, Golden E, Weir T J, Wynn A and Johnson L M 2019 Advanced fabrication processes for superconductor electronics: current status and new developments *IEEE Trans. Appl. Supercond.* **29** 1–13
[36] Whiteley S R 2022 WRspice reference manual. Whiteley research inc (available at: http://www.wrcad.com/manual/wrsmanual.pdf)
[37] Filippov T V, Sahu A, ErenÃ‡elik M, Kirichenko D E, Habib M and Gupta D 2021 Gray zone and threshold current measurements of the Josephson balanced comparator *IEEE Trans. Appl. Supercond.* **31** 1–7
[38] Semenov V 2003 Digital SQUIDs: new definitions and results *IEEE Trans. Appl. Supercond.* **13** 747–50
[39] Krylov G and Friedman E G 2017 Design for testability of SFQ circuits *IEEE Trans. Appl. Supercond.* **27** 1–7
[40] Rylov S V 2019 Clockless dynamic SFQ and gate with high input skew tolerance *IEEE Trans. Appl. Supercond.* **29** 1–5

[41] Krylov G and Friedman E G 2020 Asynchronous dynamic single flux quantum majority gates *IEEE Trans. Appl. Supercond.* **30** 1–7
[42] Jabbari T, Krylov G, Kawa J and Friedman E G 2021 Splitter trees in single flux quantum circuits *IEEE Trans. Appl. Supercond.* **31** 1–6
[43] Katam N, Shafaei A and Pedram M 2017 Design of multiple fanout clock distribution network for rapid single flux quantum technology *Proc. Asia and South Pacific Design Automation Conf.* (*ASP-DAC*) pp 384–9
[44] Krylov G and Friedman E G 2020 Design methodology for distributed large scale ERSFQ bias networks *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.* **28** 2438–47