

Spiking Neural Networks for Energy-Efficient RIS Phase Configuration

Guillaume Marthe

CITI, EA3720

Univ Lyon, INSA Lyon, Inria

69621 Villeurbanne, France

guillaume.marthe@insa-lyon.fr

Claire Goursaud

CITI, EA3720

Univ Lyon, INSA Lyon, Inria

69621 Villeurbanne, France

claire.goursaud@insa-lyon.fr

Abstract—Reconfigurable Intelligent Surfaces (RIS) need fast and energy-efficient phase configuration algorithms, especially when the controller is placed on the RIS hardware. In this paper, we investigate the use of spiking neural networks (SNNs) as an alternative to classical artificial neural networks (ANNs) for passive RIS phase optimization. Building on the RISnet architecture, we compare three configurations: the RISnet original ANN (A-A), a post-training ANN-to-SNN conversion (A-S), and a natively trained SNN using surrogate gradients (S-S). We show that the S-S configuration matches the ANN weighted sum-rate across all scenarios, while the A-S conversion is limited to moderate performances due to the analog skip connection inherent to the RISnet architecture. Energy analysis is based on arithmetic operation counts and reveals an important trade-off governed by the number of timesteps. These results demonstrate that natively trained SNNs are a promising approach for energy-efficient RIS configuration.

Index Terms—Reconfigurable Intelligent Surface, Spiking Neural Network, Energy Efficiency

I. INTRODUCTION

Future wireless networks face significant challenges regarding path loss and signal blockage, especially in non line of sight (nLOS) conditions. Conventional solutions often rely on active relays or power-intensive amplification to maintain connectivity. Reconfigurable Intelligent Surfaces (RIS) have emerged as a promising technology to address these propagation limitations and signal blockage constraints [1]. An RIS consists of an array of passive elements where each of these elements can be individually configured to shift the phase of the incident signal. This allows for precise control of the radio environment and improved signal propagation without the hardware complexity and power consumption of active systems. However, while the RIS is a passive device, the controller used to configure it is not, and the algorithms used to optimize the phase shift represent a large portion of the power consumption of the system that includes it [2]. In some applications like energy-harvested RIS or large-scale RIS deployment, the controller's consumption can become an issue [3], [4].

The most common phase optimization algorithms, such as Alternating Optimization [5], often require iterative loops

whose complexity increases with the number of RIS elements N , making them difficult to deploy in environments where power consumption may be critical. Deep learning models have been proposed to replace iterations by mapping input data in RIS configuration during a single inference step [6]. However, these algorithms are computed on artificial neural networks, requiring a large number of operations and consuming significant energy at the controller level.

Meanwhile, spiking Neural Networks (SNN) are considered as the third generation of ANNs [7]. They mimic the human brain computation by using voltage spikes as data which allows sparse and low power computations. Recent advances in SNNs, such as surrogate gradient methods [8] and the development of neuromorphic hardware like Loihi [9], have made them easier to use. Moreover, fully analog and bio-inspired hardware are being studied to better match order of energy consumption of the human brain, allowing less complex but almost passive computation [10]. The development of frameworks such as snntorch and spikingjelly [11] has also made it possible to simulate SNN training on traditional machines. As a result, SNNs appear to be an energy-efficient choice for RIS controllers with energy constraints, such as in autonomous deployments.

SNN applications for wireless communications are still rare but rising [12]. To the best of our knowledge, NeuroRIS [13] is the only SNN implementation for RIS configuration. This is mainly because RIS applications are still very recent, and usually working with SNNs implies a global rethinking of the algorithm paradigms and the data. In their work, they used a semi-passive RIS where each RIS element is connected to a neuromorphic processor which computes the desired phase shift for this element. However, they consider an ambitious dedicated digital neuromorphic RIS architecture that has yet to be demonstrated in a physical prototype. In this paper, we rather consider existing analog hardware. We propose to convert an already existing ANN architecture using a passive RIS into an SNN by transposing ReLU activations into LIF neurons. The goal is to pave the way for the straightforward conversion of existing algorithms—which are currently the focus of active research—rather than relying on dedicated neuromorphic algorithms that, to the best of our knowledge, do not exist except for NeuroRIS [13].

This work was funded by the French government under the France 2030 ANR program “PEPR Networks of the Future” (ref. ANR-22-PEFT-0004)

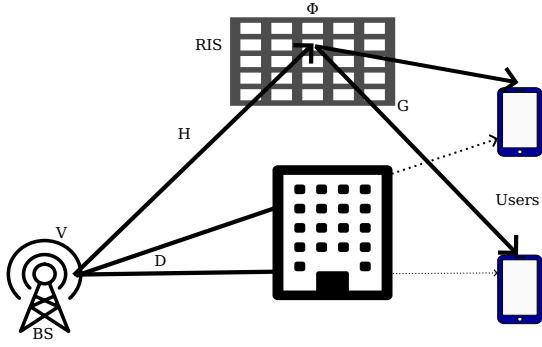


Fig. 1. System model of an RIS deployment where a building is blocking the direct path.

In particular, our objective is to use the IEMN's analog spiking neuron hardware [10] that achieves synaptic operations for only 4 fJ, which is three orders of magnitude lower than the cost of a digital accumulation on classical hardware [14]. The objective of this paper is to evaluate the performances of such a design.

To do so, we implement the RISnet architecture [6], which consider the weighted sum-rate (WSR) maximization problem, and then perform the conversion on their architecture so that we can compare performance in terms of throughput and power consumption.

The paper is organized as follows. Section II presents the system model and the problem formulation. Section III describes the proposed framework with the considered neural networks architecture as well as the energy consumption analysis. Simulation results are then studied in Section IV. Finally, Section V concludes the paper.

II. SYSTEM MODEL

A. Problem formulation

We use the RISnet architecture [6], i.e. we consider an RIS-assisted downlink broadcast channel comprising a base station (BS) equipped with M antennas serving U single-antenna users. The RIS is composed of N passive elements. The BS performs weighted minimum mean-squared error (WMMSE) precoding to serve multiple users via spatial division multiple access (SDMA). The RIS controller is placed directly on the RIS hardware, working as a low-power neuromorphic processor that computes the optimal phase configuration locally from the estimated channel state information (CSI).

The received signal at the users is:

$$\mathbf{y} = (\mathbf{G}\Phi\mathbf{H} + \mathbf{D})\mathbf{V}\mathbf{x} + \mathbf{n}, \quad (1)$$

where $\mathbf{H}$ of size $N \times M$ is the BS-to-RIS channel, $\mathbf{G}$ of size $U \times N$ is the RIS-to-users channel and $\mathbf{D}$ of size $U \times M$ is the direct BS-to-users channel. $\Phi = \text{diag}(e^{j\psi_1}, \dots, e^{j\psi_N})$ is the RIS phase shift matrix of size $N \times N$, where $\psi_n \in [0, 2\pi[$ is the phase shift of the RIS element n . A scheme of the system can be found in Fig. 1.

$\mathbf{V}$ of size $M \times U$ is the precoding matrix, $\mathbf{x}$ is the transmitted symbol vector, and $\mathbf{n}$ is the additive white Gaussian noise.

With $\mathbf{C}$ the precoding and transmission matrix :

$$\mathbf{C} = (\mathbf{G}\Phi\mathbf{H} + \mathbf{D})\mathbf{V}, \quad (2)$$

and c_{uv} the element of $\mathbf{C}$ in row u and column v , our objective is to maximize the weighted sum-rate. The WSR is defined as :

$$\text{WSR} = \sum_{u=1}^U \alpha_u \log_2 (1 + \text{SINR}_u) \quad (3)$$

where α_u is the weight of user u . The problem is then formulated as :

$$\max_{\mathbf{V}, \Phi} \sum_{u=1}^U \alpha_u \log_2 \left(1 + \frac{c_{uu}^2}{\sum_{v \neq u} c_{uv}^2 + \frac{1}{\rho}} \right), \quad (4)$$

subject to $\text{tr}(\mathbf{V}\mathbf{V}^H) \leq E_{\text{Tr}}$, $|\phi_{nn}| = 1$ for all n and $|\phi_{nn'}| = 0$ for $n \neq n'$, where E_{Tr} is the transmit power constraint and ρ is the transmit signal-to-noise ratio (TSNR). The TSNR is the ratio between transmit power and noise power. In this work, the WMMSE precoding matrix is used as $\mathbf{V}$ and Φ is optimized to maximize the WSR.

B. Channel features

In [6], Φ is obtained with a neural network depending on the channel matrices, where $\mathbf{H}$ is assumed to be constant as the BS and the RIS are stationary and the environment is considered invariant. $\mathbf{G}$ and $\mathbf{D}$ change for each user properties and are needed to obtain Φ and $\mathbf{V}$. The n -th column of $\mathbf{G}$ contains the channel between the RIS element n and the U users. $\mathbf{D}$ is the direct channel and so, does not depend on the RIS. By defining $\mathbf{J} = \mathbf{D}\mathbf{H}^+$, where $(\cdot)^+$ is the pseudo-inverse operation, (1) becomes :

$$\mathbf{y} = (\mathbf{G}\Phi + \mathbf{J})\mathbf{H}\mathbf{V}\mathbf{x} + \mathbf{n} \quad (5)$$

$\mathbf{G}$ and $\mathbf{J}$ have N columns that corresponds to the N RIS elements. The features of the RIS element n can be defined as :

$$\gamma_n = (|g_{1n}|, \arg(g_{1n}), \dots, |g_{Un}|, \arg(g_{Un}), \quad (6)$$

$$|j_{1n}|, \arg(j_{1n}), \dots, |j_{Un}|, \arg(j_{Un})) \quad (7)$$

This leads to a total of $4U$ features in the case of U users.

III. PROPOSED FRAMEWORK

A. The RISnet architecture

From the work of Peng and al. we adopt the permutation-variant RISnet architecture [6] that exploits two key properties of the RIS optimization problem. The first one is the homogeneity of the RIS elements, which means that every antenna has the same signal processing method. The second property is the decomposition of the channel information into local and global components. According to channel features γ_n defined in section II, the input of the network is the feature matrix Γ of size $4U \times N$, where each column corresponds to

one specific RIS element. RISnet architecture is composed of $L = 8$ layers. Each element is processed independently, while sharing weights to all N elements. For each layer i , the output is :

$$\mathbf{F}_{i+1} = [\Gamma \mid \mathbf{F}_{i+1}^l \mid \mathbf{F}_{i+1}^g], \quad (8)$$

where Γ is the channel feature reinjected as a skip connection at every layer. $\mathbf{F}_{i+1}^l = \text{ReLU}(\mathbf{W}_i^l \mathbf{F}_i + \mathbf{b}_i^l)$ computes the local component for each RIS element, while $\mathbf{F}_{i+1}^g = \frac{1}{N} \sum_{n=1}^N \text{ReLU}(\mathbf{W}_i^g \mathbf{F}_i + \mathbf{b}_i^g)$ computes the global component across all the elements. The final layer returns the phase shift ψ_n , comprised in $[-\pi, \pi]$. The weight sharing architecture makes the number of trainable parameters independent of N , as the computation of the $\mathbf{W}$ and $\mathbf{b}$ matrix does not rely on N .

The training phase consists in an alternating optimization procedure where at each epoch we compute the precoder $\mathbf{V}$ for the current phase configuration. We then update via gradient ascent on the WSR (4) while setting $\mathbf{V}$. This process is repeated until convergence or a set number of epochs, whichever is the fastest.

B. Spiking neural networks

Recent research often focus on the implementation of SNNs on platforms such as digital neuromorphic architectures [13]. However, dedicated analog circuits are beginning to be deployed in a few research application [12]. These ultra-low-power hardware solutions, such as the IEMN's neuron [10], consume multiple orders of energy fewer than the digital neuromorphic circuits, lowering even more the SNN consumption. Their neurons have a limited design allowing only simple neuron models, but have already proven that it is possible to reproduce a classical LIF model [12].

The RISnet architecture [6] relies on conventional ANN with ReLU activations, which can be replaced in the layers 1 to 7 by a Leaky Integrate and Fire (LIF) neuron [15]. The membrane potential of a LIF neuron varies over discrete timesteps as :

$$v_i^{(t)} = \beta v_i^{(t-1)} + \mathbf{W}_i \mathbf{x}_i^{(t)}, \quad (9)$$

with $\beta \in (0, 1)$ is the trainable decay factor of the neuron, and $\mathbf{W}_i$ are the synaptic weights. When the membrane potential reaches its threshold v_{th} , the neuron produces a spike and sends it to the next neurons while its own potential is reset to the resting potential v_{rest} .

The principle is well known nowadays for feed-forward networks [15], a ReLU activation value can be approximated by the firing rate of a spiking neuron over T timesteps, and the matching improves as T rises. In a rate-coding SNN, the data is converted as a spike train, where its value is proportional to the spike rate. T determines the number of timesteps over which the input is presented, acting as a quantization factor for the input representation.

According to this new spiking paradigm, we need to rethink the training phase. In this work we will investigate on whether it is more optimal to train the network as an ANN and use a weight-transfer approach to transform it as an SNN, or to convert it as a naive SNN, and perform directly the training on this new network. We note A-A the ANN trained as an ANN. We will then study an SNN trained on an ANN, referred to as A-S and an SNN trained directly on an SNN, noted S-S.

C. ANN to SNN conversion (A-S)

We first consider an ANN to SNN conversion as a neuro-inspired alternative to the ANN. The RISnet architecture contains skip connections in (8) that reinject continuous values of the feature matrix Γ alongside the binary spikes at each layer. This is because the channel information requires a precision that we cannot reach with rate-coding without using a high number of timesteps to quantize the data properly. This creates a mix of analog and digital data which is an issue because standard conversion methods requires a purely spike driven data [15]. We evaluated a few conversion methods like weight normalization or full spike encoding of Γ , but the best results were obtained with Dhiel's weight transfer approach [15]. We use the direct weight transfer to LIF neurons with a decay rate of $\beta = 0.9$, and $T = 32$ timesteps.

D. SNN architecture (S-S)

In addition we propose an SNN architecture inspired by the RISnet [6] architecture. The spikes replace the ReLU output, and the feature matrix Γ is still reinjected as an analog skip connection :

$$\mathbf{F}_{i+1}^{(t)} = [\Gamma \mid \mathbf{s}_{i+1}^{l,(t)} \mid \mathbf{s}_{i+1}^{g,(t)}]. \quad (10)$$

The last layer is still a classical linear readout, and the phase output is obtained by averaging over T timesteps :

$$\psi = \frac{\pi}{T} \sum_{t=1}^T \mathbf{W}_8 \mathbf{F}_8^{(t)}. \quad (11)$$

The network is trained with a surrogate gradient method because the spike function is non differentiable. Thus, it is approximated with a fast sigmoid during backpropagation. The AO procedure, the loss function and optimizer are the same as the ANN.

E. Energy Consumption

We compute the inference energy by counting arithmetic operations. For the ANN, every synaptic connection performs a multiply accumulate (MAC) operation. The energy for the l -th layer of the ANN with d_{in} input channels and d_{out} output channels is :

$$E_{ANN}^{(l)} = d_{in} \cdot d_{out} \cdot N \cdot \varepsilon_{MAC}, \quad (12)$$

with $\varepsilon_{MAC} = 4.6$ pJ for a 32-bit floating-point MAC at 45 nm [14].

One of the key features of RISnet is the analog skip connection. In the SNN, the ReLU activations are replaced

by LIF neurons but as the skip connection is not translated into spikes, it still needs MAC operations at each timestep because the features are still continuous values.

The total ANN energy becomes:

$$E_{\text{ANN}}^{\text{total}} = E_{\text{ANN}}^{\Gamma} + E_{\text{ANN}}^{\text{spike}} \quad (13)$$

where E_{ANN}^{Γ} the skip connection part of the energy for the ANN and the SNN, evaluated by counting the MAC operations involving Γ at each layer is :

$$E_{\text{ANN}}^{\Gamma} = d_{\Gamma} \cdot d_{\text{out}} \cdot L \cdot N \cdot \varepsilon_{\text{MAC}} \quad (14)$$

where $d_{\Gamma} = 4U$ as seen in section II and the ANN's layers part is :

$$E_{\text{ANN}}^{\text{spike}} = \sum_{l=1}^L E_{\text{ANN}}^{(l)} \quad (15)$$

In an SNN, a computation occurs when a presynaptic neuron fires a spike. Since the spike is binary, the operation on digital hardware is only an accumulation (AC). The energy also depends on the measured spike rate $\rho_l^{(t)}$:

$$E_{\text{SNN}}^{(l)} = \sum_{t=1}^T \rho_l^{(t)} \cdot d_{\text{in}} \cdot d_{\text{out}} \cdot N \cdot \varepsilon_{\text{AC}_X}, \quad (16)$$

with $\varepsilon_{\text{AC}_X} = \varepsilon_{\text{AC}_D} = 0.9$ pJ [14] for a digital platform and $\varepsilon_{\text{AC}_X} = \varepsilon_{\text{AC}_I} = 4$ fJ for the IEMN's neuron.

The SNN total energy consumption is :

$$E_{\text{SNN}}^{\text{total}} = T \cdot E_{\text{ANN}}^{\Gamma} + E_{\text{SNN}}^{\text{spike}} \quad (17)$$

where the SNN's spiking energy consumption $E_{\text{SNN}}^{\text{spike}}$ is :

$$E_{\text{SNN}}^{\text{spike}} = \sum_{l=1}^L E_{\text{SNN}}^{(l)} \quad (18)$$

The T factor appears because the product $\mathbf{W}_{\gamma}\Gamma$ will be computed during T timesteps on digital processor to allow for a T times smaller quantization. As the skip connection energy depends on the deployment hardware, we will use three metrics. The first one is the full energy ratio including E_{ANN}^{Γ} :

$$\eta^{\text{full}} = \frac{E_{\text{ANN}}^{\text{total}}}{E_{\text{SNN}}^{\text{total}}} \quad (19)$$

The second one measures the computations independently of how the skip-connection is reinjected in the network and considering a digital platform:

$$\eta_D^{\text{spike}} = \frac{E_{\text{ANN}}^{\text{spike}}}{E_{\text{SNN}}^{\text{spike}}} \quad (20)$$

The third metric $\eta_{\text{IEMN}}^{\text{spike}}$ is measured while making the assumption that we use the IEMN neuron and the memristive conductances from [16] and $\eta_{\text{IEMN}}^{\text{spike}}$ has the same expression as η_D^{spike} :

$$\eta_{\text{IEMN}}^{\text{spike}} = \frac{E_{\text{ANN}}^{\text{spike}}}{E_{\text{IEMN}}^{\text{spike}}} \quad (21)$$

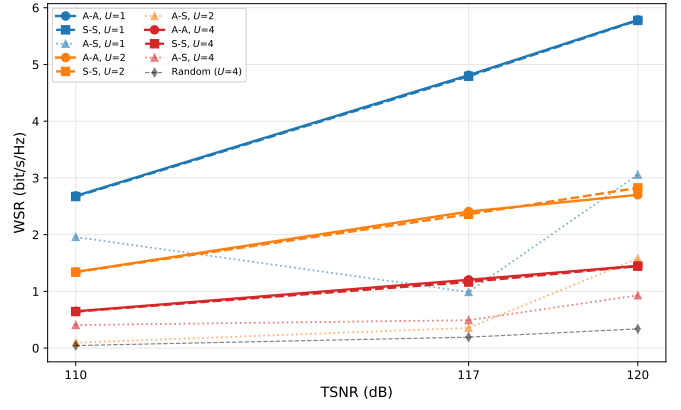


Fig. 2. Weighted sum-rate depending on the TSNR for $U = 1, 2$, and 4 users.

IV. SIMULATION RESULTS

A. Setup

We use the ray-tracing channel dataset from [6] with $N = 1024$ RIS elements, $M = 9$ antennas and an equal weight of all users $\alpha_u = 1/U$. Three multi-users scenarios are evaluated : $U \in \{1, 2, 4\}$. The training set contains 10240 individual channel samples organized in 5120 groups of U users, and the test set contains 1024 groups. The same three TSNR values from RISnet are considered : 110 dB, 117 dB and 120 dB. Both the ANN and the SNN architectures contain $L = 8$ layers. The S-S configuration is tested with $T = \{1, 2, 4, 8\}$ timesteps and is implemented with the snntorch library [17]. The A-S configuration uses $T = 32$ with direct weight transfer [15] from the trained ANN to obtain better performances. All the models uses the Adam optimizer using alternating optimization with WMMSE precoding. An early stopping is applied based on the WSR convergence. A different model is trained for each pair of U and TSNR.

B. WSR performance

Fig. 2 represents the WSR achieved after training the three architectures for three values of U and TSNR. We also plotted the random case for $U = 4$ as a lower bound. It is evaluated by drawing random phase-shifts uniformly from $[0, 2\pi)$ for every RIS element instead of using optimized phase-shifts. We compared our results and observed that the performances of our ANN matches the performances of the RISnet architecture [6], validating our implementation.

We can observe that for every cases, the S-S configuration almost matches the ANN performances with between 98.3% to 99.8% of the WSR. On the other hand, the A-S architecture only achieves 70% to 80% of the performances of the A-A architecture, while we use $T = 32$ to achieve better performance at the cost of higher energy consumption. This is because the conversion method requires a purely spike-driven network, which is impossible due to the issue with Γ . The performance gap is narrowing as U and TSNR increase, but they remain insufficient when compared to S-S in the figure.

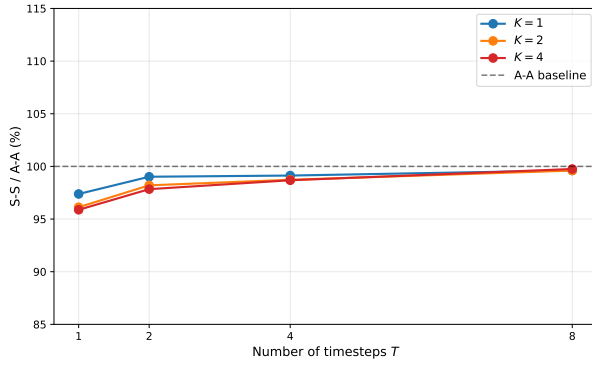


Fig. 3. Weighted sum-rate depending on T , for $T = 1, 2, 4$ and 8 timesteps.

This results shows that the training directly on the SNN with surrogate gradient method is more well suited to this problem than post-training conversion.

C. WSR vs T

Fig. 3 represents the evolution of the performances based on the number of timesteps T for multiple number of users U . It can also be seen as the quantization permitted for each input data. We can see that the three curves share the same evolution. For the same number of users U , the performances tend to improve if we increase T as the network has a better information on the channel. Following the same logic, for a set T value, increasing the number of users U tends to decrease the performances, this is particularly true for lower values of T , whereas for higher values, T is long enough so that the impact of computing more users is attenuated.

D. Energy analysis

We plotted the energy ratios between the S-S and A-A architectures using the three metrics defined in section III-D, averaged on the three values of TSNR. The full architecture energy ratio η^{full} (19) (red) is measuring the energy cost of the Γ skip connection in addition to the spike-driven layers. On a digital platform like Loihi [9], the SNN consume 1.3 to 3.2 times more energy than the corresponding ANN. Γ 's dimension grows when U increases, while the dimension of the other inputs remain constant.

We can see that the more U increases, the more features there are ($4U$ features), the more the MAC operations dominate the total cost. This is because of Γ being processed again at each of the $T = 8$ timesteps.

The spike-only ratio η_D^{spike} (20) (blue) measures the energy cost of the spike driven layers, using $\varepsilon_{AC} = 0.9$ pJ per AC [14]. With this metric, the SNN consumes 2.3 to 3.3 times less energy than the equivalent ANN layers, with $\varepsilon_{MAC} = 4.6$ pJ. This gain is mainly due to the trained network sparsity where measured spike rate are between 8 and 39% depending on the layer exceeding the losses due to the multiple timesteps.

Finally, we plotted the spike-only energy ratio using the ultra-low-power (ULP) neuron circuit presented in [10] which

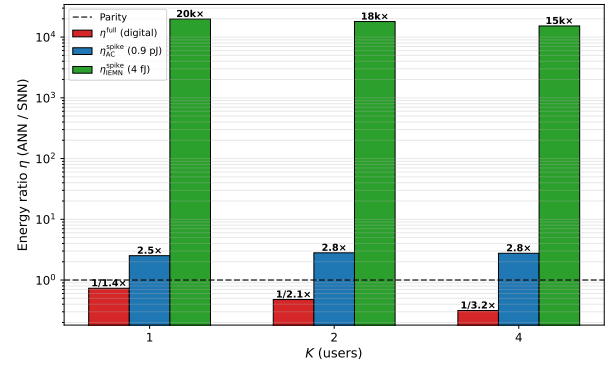


Fig. 4. Energy comparison between the A-A and S-S configurations.

only consumes 4fJ per spike. Using this hardware, the S-S configuration achieved to obtain a ratio $\eta_{IEMN}^{\text{spike}}$ (green) higher than 15.000, highlighting the potential of emergent neuromorphic hardware for tasks like RIS configuration.

These results illustrate that the energy efficiency of the digital implementation of the SNN is currently tied to the implementation of the skip connection because the consumption of the computation part of the network is becoming negligible. Using a constant analog signal such as a memristive crossbar array [16] would make the spike-only ratio become the more realistic consumption model, and the SNN would offer a very relevant energy-efficient solution to the RIS configuration problem.

In NeuroRIS [13], for the same hardware and considering $\eta_{IEMN}^{\text{spike}}$, their ratio between the ANN and SNN consumption's seems to be around $\eta_{IEMN}^{\text{spike}} = 100$ to 1000 . The IEMN neuron thus permits a lower energy consumption.

Finally, we plotted on Fig. 5 the performances of multiple studied configuration, depending on the consumption of this configuration. Both metrics are evaluated as a ratio between the metric for the considered SNN and the metric for the corresponding ANN. The right part, colored in green, is the space where the consumption of the SNN is lower than the one from the ANN. The leftmost cluster represent the configuration if we consider η^{full} . The rightmost correspond to the same configurations considering $\eta_{IEMN}^{\text{spike}} \cdot \eta_D^{\text{spike}}$ was not displayed on this Figure for clarity, and would have been placed the same way as η^{full} , but shifted half a decade to the right. The main tendency for η^{full} curves is that for $T = 8$, all models match the performances of the ANN for at least 98%, but they consumes more energy than the ANN, because we input the data during 8 timesteps instead of only once. On the other hand, for $T = 1$, the performances are between 92% and 98%, but consumes between 4 to 8 times less than the corresponding ANN for the η^{full} metric. For the in-between values $T = 2$ and $T = 4$ performances and consumption follow the tendency, when T increase the performances and the consumption all increase, showing that there is trade-off to study to choose T optimally, depending on the metrics we want to optimize the most. Finally, this graph highlights the

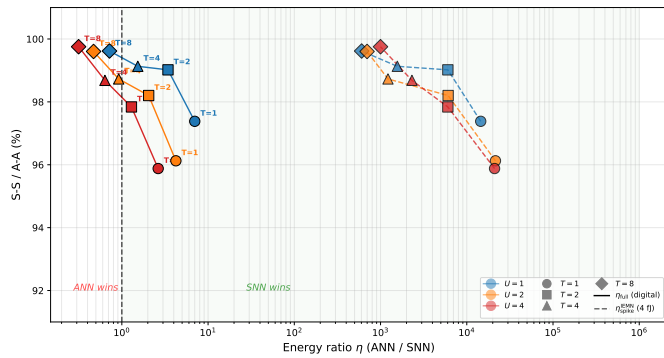


Fig. 5. Summary of the performances and the energy consumption of multiple SNN configurations compared to the ANN.

ultra-low-power consumption of the IEMN's neuron, as this architecture consumes between 3 and 5 orders of magnitude less energy than an ANN architecture, and between 10 and 100 times less than NeuroRIS [13], confirming the interest of these emergent neuromorphic dedicated circuit for energy efficient RIS configuration.

V. CONCLUSION

In this paper, we discussed the use of spiking neural networks to realize an energy efficient RIS configuration by adapting the RISnet architecture to the spike domain. In this study, we trained and tested three different neural network configurations (A-A, A-S and S-S) and evaluated their performances in terms of WSR and energy consumption. Initially, we observed that training on a naive network was more effective than post-training conversion. The S-S achieves the same performance as the A-A simply by replacing the ReLU layers with LIF neurons, while consuming less energy for the spike-driven part, which is a very encouraging result. Since the conditions for conversion are not met for the A-S, it performs worse and consumes more power than the other two. We observed that for this rate-coding application, if the network has more time to read the data, which means T is high, the performances are enhanced. However, lowering this duration reduces the energy cost substantially, without decreasing the performances that much. For ultra-low-power constraints, it could be interesting to evaluate this trade-off more carefully.

The IEMN's neuron appears as an ultra low power computing solution for RIS configuration compared to digital hardwares. However, these analog circuits are made by a foundry process making its functionality fixed, unlike reprogrammable digital platforms such as Loihi [9]. This implies that the model needs to be validated before deployment, which is compatible with the offline learning used in this work.

Finally, it would be interesting to compare performance with an SNN network that uses a time-based input rather than a rate-based one. This would require changing the entire architecture, but a time-based approach results in fewer spikes for the same computation, and thus consumes less energy for the same task.

However, these solutions are often more complex to implement and cannot be achieved through a simple conversion; this trade-off will be examined.

REFERENCES

- [1] M. D. Renzo, A. Zappone, M. Debbah, M. Alouini, C. Yuen, J. de Rosny, and S. A. Tretyakov, "Smart radio environments empowered by reconfigurable intelligent surfaces: How it works, state of research, and road ahead," *CoRR*, vol. abs/2004.09352, 2020. [Online]. Available: <https://arxiv.org/abs/2004.09352>
- [2] A. Ghaneizadeh, P. Gavrilidis, M. Joodaki, and G. C. Alexandropoulos, "Metasurface energy harvesters: State-of-the-art designs and their potential for energy sustainable reconfigurable intelligent surfaces," 2024. [Online]. Available: <https://arxiv.org/abs/2407.14638>
- [3] S. Bansal, V. K. Malav, and A. Sharma, "Battery-less reconfigurable intelligent surface powered by energy harvesting rectenna array," in *2024 IEEE Microwaves, Antennas, and Propagation Conference (MAPCON)*, 2024, pp. 1–5.
- [4] S. Lyu, L. Peng, and S. Y. Chang, "Investigating large-scale ris-assisted wireless communications using gnn," *IEEE Transactions on Consumer Electronics*, vol. 70, no. 1, pp. 811–818, 2024.
- [5] N. S. Perović, L.-N. Tran, M. D. Renzo, and M. F. Flanagan, "On the maximum achievable sum-rate of the ris-aided mimo broadcast channel," *IEEE Transactions on Signal Processing*, vol. 70, pp. 6316–6331, 2022.
- [6] B. Peng, F. Siegmund-Poschmann, and E. A. Jorswieck, "RISNet: A dedicated scalable neural network architecture for optimization of reconfigurable intelligent surfaces," in *Proc. IEEE WSA & SCC*, 2023.
- [7] W. Maass, "Networks of spiking neurons: The third generation of neural network models," *Neural Networks*, vol. 10, no. 9, pp. 1659–1671, 1997.
- [8] X. Yao, Q. Hu, F. Zhou, T. Liu, Z. Mo, Z. Zhu, Z. Zhuge, and J. Cheng, "Spinerf: direct-trained spiking neural networks for efficient neural radiance field rendering," *Frontiers in Neuroscience*, vol. Volume 19 - 2025, 2025.
- [9] M. Davies, N. Srinivasa, T.-H. Lin, G. Chinya, Y. Cao, S. H. Choday, G. Dimou, P. Joshi, N. Imam, S. Jain, Y. Liao, C.-K. Lin, A. Lines, R. Liu, D. Mathaikutty, S. McCoy, A. Paul, J. Tse, G. Venkataramanan, Y.-H. Weng, A. Wild, Y. Yang, and H. Wang, "Loihi: A neuromorphic manycore processor with on-chip learning," *IEEE Micro*, vol. 38, no. 1, pp. 82–99, 2018.
- [10] I. Sourikopoulos, S. Hedayat, C. Loyez, F. Danneville, V. Hoel, E. Mercier, and A. Cappy, "A 4-fj/spike artificial neuron in 65 nm cmos technology," *Frontiers in Neuroscience*. [Online]. Available: <https://www.frontiersin.org/article/10.3389/fnins.2017.00123>
- [11] W. Fang, Y. Chen, J. Ding, Z. Yu, T. Masquelier, D. Chen, L. Huang, H. Zhou, G. Li, and Y. Tian, "Spikingjelly: An open-source machine learning infrastructure platform for spike-based intelligence," 2023. [Online]. Available: <https://arxiv.org/abs/2310.16620>
- [12] G. Marthe, C. Goursaud, and L. Clavier, "Wake-up radio receiver based on spiking neurons for detecting activation sequence," in *2023 IEEE Wireless Communications and Networking Conference*.
- [13] C. G. Tsinos, A.-A. A. Bouligeorgos, and T. A. Tsiftsis, "Neuroris: Neuromorphic-inspired metasurfaces," 2023. [Online]. Available: <https://arxiv.org/abs/2312.14548>
- [14] M. Horowitz, "1.1 computing's energy problem (and what we can do about it)," in *2014 IEEE International Solid-State Circuits Conference Digest of Technical Papers (ISSCC)*, 2014, pp. 10–14.
- [15] P. U. Diehl, D. Neil, J. Binas, M. Cook, S.-C. Liu, and M. Pfeiffer, "Fast-classifying, high-accuracy spiking deep networks through weight and threshold balancing," in *Proc. IEEE Int. Joint Conf. Neural Netw. (IJCNN)*, 2015, pp. 1–8.
- [16] F. Nowshin, H. An, and Y. Yi, "Towards energy-efficient spiking neural networks: A robust hybrid cmos-memristive accelerator," *ACM J. Emerg. Technol. Comput. Syst.*, vol. 20, no. 1, pp. 1–20, 2024.
- [17] J. K. Eshraghian, M. Ward, E. Neftci, X. Wang, G. Lenz, G. Dwivedi, M. Bennaamoun, D. S. Jeong, and W. D. Lu, "Training spiking neural networks using lessons from deep learning," *Proceedings of the IEEE*, vol. 111, no. 9, pp. 1016–1054, 2023.