Spiking Sparse Recovery With Non-Convex Penalties

Xiang Zhang, Lei Yu, *Member, IEEE*, Gang Zheng, *Senior Member, IEEE*, and Yonina C. Eldar, *Fellow, IEEE*

***Abstract*—Sparse recovery (SR) based on spiking neural networks has been shown to be computationally efficient with ultra-low power consumption. However, existing spiking-based sparse recovery (SSR) algorithms are designed for the convex ℓ_1-norm regularized SR problem, which often underestimates the true solution. This paper proposes an adaptive version of SSR, *i.e.*, A-SSR, to optimize a class of non-convex regularized SR problems and analyze its global asymptotic convergence. The superiority of A-SSR is validated with synthetic simulations and real applications, including image reconstruction and face recognition. Furthermore, it is shown that the proposed A-SSR essentially improves the recovery accuracy by avoiding systematic underestimation and obtains over 4 dB PSNR improvement in image reconstruction quality and around 5% improvement in recognition confidence. At the same time, the proposed A-SSR maintains energy efficiency in hardware implementation. When implemented on the neuromorphic Loihi chip, our method consumes only about 1% of the power of the iterative solver FISTA, enabling applications under energy-constrained scenarios.**

***Index Terms*—Sparse recovery, spiking neural network, non-convex optimization.**

I. Introduction

SPARSE recovery (SR) is an essential problem in many applications of signal and image processing [1]. Direct optimization of sparse regularized problems, *e.g.*, iterative shrinkage-thresholding algorithm (ISTA) [2], basis pursuit denoising (BPDN) [3] and least absolute shrinkage and selection operator (LASSO) [4], often suffer from heavy computational burden and power inefficiency due to the necessity of iterative numerical methods to guarantee convergence, thus limiting applications in real-world scenarios. Inspired by the efficient communication scheme observed in biological nervous systems, artificial spiking neurons transmit information with other neurons via sparsely emitted spikes [5], leading to extremely low power consumption of spiking neural networks (SNNs). The computability of SNNs has been proven in the pioneering works of [6], [7]. Recent works have also established the efficiency of SNNs in solving specific problems such as dictionary learning [8] and SR [9], [10], [11].

In the realm of SR, many works have focused on developing SNN-based algorithms. Zylberberg et al. first demonstrated that SR principles can operate on spiking neurons through synaptically local plasticity rules [12], but they did not provide a specific energy function. Hu et al. proposed a hybrid distributed algorithm (HDA), which is equivalent to a network of spiking neurons, to solve a time-varying LASSO problem [13]. Compared to other distributed algorithms, HDA shows important advantages such as energy efficiency and low bandwidth requirements. Exploiting the sparse nature of the output, the total power of SNNs is further reduced when dealing with SR problems [9], [10]. Shapero et al. were the first to implement the spiking SR (SSR) algorithm on a Field Programmable Analog Array (FPAA) chip [9]. Though the SSR algorithm is restricted to non-negative variables to mimic biophysical variables (e.g., firing rates of spiking neurons), their experiments still validate the extremely high efficiency and low power consumption of SSR systems, demonstrating huge potential in practical usage.

Most existing SSR algorithms are designed to solve the convex ℓ_1-norm regularized optimization problem, which tends to underestimate high-amplitude components and often results in biased estimates. On some settings, the result of ℓ_1-norm regularized optimization is not sparse enough, or the original signal cannot be reliably recovered [14]. In contrast, numerous researches have shown that non-convex regularized optimization can lead to more accurate estimates since non-convex penalties do not penalize high-amplitude components excessively [14], [15]. Related algorithms can be roughly divided into three classes. The first class is the iterative reweighted algorithms, including the iterative reweighted least squares (IRLS) [16], iterative reweighted ℓ_1 minimization (IRL1) [14] and their extended versions for the ℓ_q-norm $(0 < q < 1)$ regularized minimization problems [17]. These algorithms improve the recovery performance via solving a sequence of weighted minimization problems where the weights used for the next iteration are updated according to the current solution. The second class is the difference of convex functions algorithm (DC programming) [15], which is based on an iterative algorithm that solves a convex weighted LASSO problem at each iteration. The last class is iterative thresholding algorithms [18]. By designing a thresholding operator corresponding to the chosen non-convex penalty, iterative thresholding is effective and easy to implement when dealing with large scale problems [19].

Manuscript received 8 February 2022; accepted 30 December 2022. Date of publication 5 January 2023; date of current version 19 January 2023. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Gongguo Tang. This work was supported in part by the National Natural Science Foundation of China under Grants 61871297 and 62271354, and in part by the Natural Science Foundation of Hubei Province, China under Grant 2021CFB467. *(Corresponding author: Lei Yu.)*

Xiang Zhang and Lei Yu are with the School of Electronic and Information, Wuhan University, Wuhan 430072, China (e-mail: xiangz@whu.edu.cn; ly.wd@whu.edu.cn).

Gang Zheng is with the INRIA Lille Villeneuve d'Ascq, 59650 Villeneuve d'Ascq, France (e-mail: gang.zheng@inria.fr).

Yonina C. Eldar is with the Weizmann Institute of Science, Rehovot 7610001, Israel (e-mail: yonina.eldar@weizmann.ac.il).

Digital Object Identifier 10.1109/TSP.2023.3234460

1053-587X © 2023 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission.
See https://www.ieee.org/publications/rights/index.html for more information.

Authorized licensed use limited to: Weizmann Institute of Science. Downloaded on January 22,2023 at 10:47:14 UTC from IEEE Xplore. Restrictions apply.

TABLE I
COMPARISON OF POWER CONSUMPTION

Dictionary Size ($M \times N$)	8×16	16×32	32×64	64×128	128×256	256×512	512×1024
Power of FISTA [23]	1.61W	1.76W	1.83W	1.85W	1.94W	2.41W	8.77W
Power of SSR [10]	**12.55mW**	16.34mW	19.76mW	21.40mW	27.02mW	30.61mW	36.22mW
Power of A-SSR	12.87mW	**14.30mW**	**18.44mW**	**20.71mW**	**25.21mW**	**30.15mW**	**35.85mW**
Power Ratio FISTA/A-SSR	125×	123×	99×	89×	77×	80×	244×

The result we present here is the active power, which is obtained by subtracting the background power from the total power. The lowest power consumption is marked in bold.

The performance of SR is significantly improved by exploiting non-convex penalties, but the iterative process of the above techniques hinders the overall computational efficiency. On the other hand, SNN-based algorithms have shown excellent power efficiency, motivating the combination of SNN and non-convex regularized optimization. Previous works [9], [10] show that SSR systems have equivalent dynamics to the non-spiking locally competitive algorithm (LCA) [20] when dealing with the convex ℓ_1-norm regularized SR problems. In [21], the authors present a general approach to implement LCA for a wide variety of cost functions, including non-convex ones. Combining [9] and [21] suggests replacing the convex penalties of existing SSR approaches with non-convex penalties. However, the following challenges remain unaddressed:

- *Architecture Design:* Existing SSR systems [9], [10], [11] only describe the architecture with convex ℓ_1-norm penalties. Thus, modifications should be made on SSR systems to incorporate non-convex penalties while maintaining the fundamental spiking dynamics. Although the idea of combining [9] and [21] inspires several possible solutions for implementing the SSR systems with non-convex penalties, the corresponding theoretical properties including system stability, convergence conditions, and solution equivalence, need to be considered.
- *Theoretical Analysis.* Prior work [10] provides theoretical analysis of the SSR system with convex ℓ_1-norm penalties [9]. However, this result does not hold for non-convex penalties. In addition, the theoretical properties such as system stability, solution equivalence, and convergence conditions are closely related to the specific form of the designed spiking system, which might vary with different non-convex penalties.

Inspired by the results of [9], [10], [21], this paper presents an adaptive version of SSR (A-SSR) to solve a class of non-convex regularized SR problems with SNNs. In addition, we provide theoretical analysis of A-SSR, including system stability, solution equivalence, and convergence conditions. By adapting the spiking model, the proposed method encourages the recovery of high-amplitude components, leading to improvement in recovery accuracy. At the same time, A-SSR maintains the advantages of simple hardware implementation and low power consumption. When implemented on the neuromorphic Loihi chip [22], the proposed method can solve SR problems with approximately 1% of the power consumption of fast iterative shrinkage-thresholding algorithm (FISTA) [23], as illustrated in Table I. Furthermore, we prove the global asymptotic convergence of A-SSR and show that A-SSR converges to a critical point of the corresponding non-convex regularized optimization problem, theoretically guaranteeing the system stability and recovery accuracy in practical applications. The relationship between SR algorithms and the corresponding optimization problems is summarized in Fig. 1 for clarity.

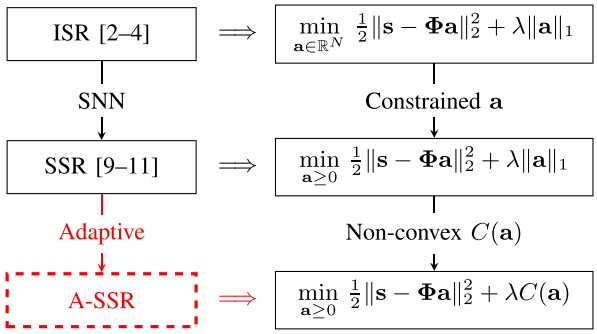


Fig. 1. Relationship between SR algorithms and optimization problems. ISR, SSR and A-SSR denote iterative, spiking and adaptive spiking sparse recovery algorithms respectively. $A \Longrightarrow B$ represents that algorithm A solves optimization problem B, and the red part indicates our contribution.

The rest of this paper is organized as follows. The basic concepts of spiking neurons and the corresponding configuration for the ℓ_1-norm regularized SR problem (*i.e.*, SSR shown in Fig. 1) are introduced in Section II. Our A-SSR approach is proposed in Section III by applying an adaptive mechanism to the spiking neuron model. The relationship between A-SSR and the non-convex regularized SR problem is established and convergence of A-SSR is analyzed. Finally, we design a series of experiments in Section IV to verify the performance of A-SSR in terms of recovery accuracy and computational efficiency, and demonstrate the superiority of A-SSR in real applications.

II. PRELIMINARY OF SSR

In this section, we first review the basic concepts of SNNs and spiking neurons, and then introduce the development of neuromorphic computing platforms designed for SNN. Finally, we describe how SNN can be configured to solve SR problems.

A. SNNs and Spiking Neurons

Instead of using real-valued computation, *i.e.*, the amplitude of the signal, SNNs exploit the timing of the signals to process information [24] in the form of binary spikes (either 1 or 0),

Authorized licensed use limited to: Weizmann Institute of Science. Downloaded on January 22,2023 at 10:47:14 UTC from IEEE Xplore. Restrictions apply.

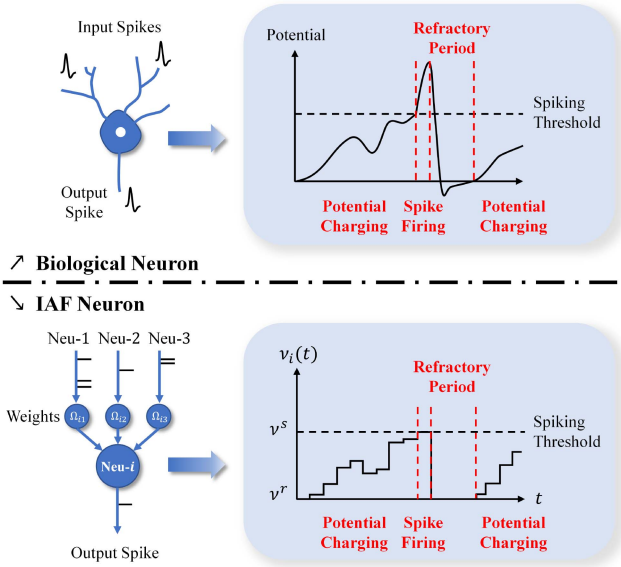


Fig. 2. Activities of biological neurons and IAF neurons include three main stages: potential charging, spike firing and refractory period.

leading to two main advantages: *computational efficiency* and *low power consumption*. At first, the mathematical dot-product operations in ANNs can be replaced by less computationally intensive binary operations in SNNs, reducing the computational burden [24]. On the other hand, spiking neurons are only activated when they receive or emit spikes, otherwise they stay in idle mode to save energy. These characteristics potentially make the spiking neuron an efficient computation unit. When encountering SR problems, the computational efficiency can be further enhanced since spiking neurons are sparsely activated [9], [25].

A variety of spiking neuron models, e.g., integrate-and-fire (IAF) [26] and Hodgkin-Huxley (HH) [27], have been proposed to describe the generation of spikes at different levels of biofidelity. In this work, we use the simple yet versatile IAF model to build SNNs, where the neuron behavior can be divided into three stages: *potential charging, spike firing* and *refractory period*, as illustrated in Fig. 2.

Potential Charging: Taking neuron-i as an example, its membrane potential $\nu_i(t)$ updates according to

$$\nu_i(t) = \int_{t_{i,l}}^{t} (\mu_i(s) + \mu^c)ds + \nu_i(t_{i,l}) \tag{1}$$

where μ^c is the constant current, $\mu_i(t)$ is the input current and $t_{i,l}$ is the latest firing time of neuron-i ($t_{i,l} = 0$ if no spike is emitted). The term 'firing time' refers to the moment when a given neuron emits a spike. Collecting all the firing times of neuron-i, we can define the spike sequence as

$$\sigma_i(t) = \sum_{k=1}^{l} \delta(t - t_{i,k}) \tag{2}$$

where $\delta(\cdot)$ is the Dirac δ function and $t_{i,k}$ denotes the k-th firing time of neuron-i.

Spike Firing: Neuron-i emits spikes when its membrane potential $\nu_i(t)$ reaches a predefined spiking threshold ν^s and then resets $\nu_i(t)$ to the resting potential $\nu^r < \nu^s$. Correspondingly, the firing time $t_{i,k}$ is defined formally as follows,

$$t_{i,k} : \ \nu_i(t_{i,k}) = \nu^s. \tag{3}$$

Combing the potential resetting mechanism with (1), we reformulate (1) as

$$\nu_i(t) = \int_0^t (\mu_i(s) + \mu^c)ds - (\nu^s - \nu^r) \int_0^t \sigma_i(s)ds \tag{4}$$

where the last term indicates the potential removed while resetting the neuron after each spike (here we assume $\nu_i(0) = 0$).

Refractory Period: After the spike firing, neuron-i immediately enters the refractory period for a given time interval $T^{ref} > 0$, during which $\nu_i(t)$ remains at ν^r and no new spike is emitted from neuron-i. Note that T^{ref} is usually set small enough that there is no noticeable effect on the network dynamics. At last, neuron-i returns to the charging period and starts updating its membrane potential $\nu_i(t)$.

To apply IAF neurons to specific tasks, the basic methodology is to configure the constant current μ^c and design the connectivity of neurons that relate to the input current $\mu_i(t)$. After running the configured SNN for a given time interval, the output of network can be obtained by encoding the spike sequence [28].

B. Neuromorphic Computing Platforms

Due to the asynchronous nature of spiking neurons, traditional von Neumann architecture cannot fully utilize the advantages of SNN [24], thus the research on neuromorphic computing platforms becomes necessary. To simulate the working mechanism of the cerebral cortex, IBM developed the neuromorphic chip TrueNorth [29] with mature CMOS integrated circuit technology. Each chip integrates 1 million programmable spiking neurons and 256 million synaptic connections, enabling it to handle large-scale problems. Inspired by the connectivity characteristics of the mammalian brain, the SpiNNaker project has been proposed [30], aiming to deliver a massively parallel million-core computer suitable for modeling large SNNs in real biological time. Using a novel event-routing technology, SynSense developed the dynamic neurormorphic asynchronous processor (DYNAP) [31], which provides an ultra-low-power and ultra-low-latency neuromorphic device for a variety of artificial intelligence edge computing applications. In addition, Intel developed a 60-mm^2 on-chip learning processor, namely Loihi [22], [32]. Each Loihi chip supports 130 thousand neurons and 2.1 million synapses per mm^2, which is the highest synapse density in current neuromorphic chips. Recently, Tsinghua University successfully developed the Tianjic chip [33], which is the first hybrid, synergistic platform. By building the hybrid architecture of artificial neural networks (ANNs) and SNNs on one chip, Tianjic provides a generalized platform for practical tasks.

Recent advances in neuromorphic computing platforms show a promising future for SNN and guarantee the performance of SNN-based algorithms. We next introduce how the SNN can be employed to solve SR problems.

Authorized licensed use limited to: Weizmann Institute of Science. Downloaded on January 22,2023 at 10:47:14 UTC from IEEE Xplore. Restrictions apply.

C. Spiking Sparse Recovery

Consider an M-dimensional stimulus $\mathbf{s} \in \mathbb{R}^M$ (*e.g.*, an M-pixel image) modeled as

$$\mathbf{s} = \mathbf{\Phi a} + \boldsymbol{\epsilon} \tag{5}$$

where $\mathbf{\Phi} = [\boldsymbol{\phi}_1, \ldots, \boldsymbol{\phi}_N] \in \mathbb{R}^{M\times N}$ with $M < N$ is the sensing matrix or dictionary and $\boldsymbol{\epsilon} \in \mathbb{R}^M$ is the noise. SR algorithms aim at recovering an N-dimensional sparse signal $\mathbf{a} \in \mathbb{R}^N$ from $\mathbf{s}$. One of the most popular methods is to solve a regularized optimization problem

$$\min_{\mathbf{a}\in\mathbb{R}^N} \frac{1}{2}\|\mathbf{s} - \mathbf{\Phi a}\|_2^2 + \lambda C(\mathbf{a}), \quad \lambda > 0, \tag{6}$$

where $C(\mathbf{a}) : \mathbb{R}^N \to \mathbb{R}_+$ is a sparsity inducing penalty. Ideally, we may choose the ℓ_0-norm as the penalty $C(\mathbf{a}) = \|\mathbf{a}\|_0$, which counts the number of nonzero elements in the sparse signal. Due to the non-convexity and NP-hard issue [34] of ℓ_0-norm regularized optimization, convex functions are often employed as surrogate penalties, especially the ℓ_1-norm penalty $C(\mathbf{a}) = \|\mathbf{a}\|_1$. With convexity of the ℓ_1-norm, the optimization problem (6) becomes tractable but traditional optimization methods [2], [3], [4] often require a large number of iterations, which is highly power consuming.

Instead of direct optimization, dynamic systems such as Hopfield-style networks [35] can be employed to solve the ℓ_1-norm regularized optimization problem. The locally competitive algorithm (LCA) [20], [21] is a typical example. Specifically, each neuron in the LCA is characterized by an internal state variable, $u_i(t)$ for $i = 1, \ldots, N$, and produces outputs, $a_i(t)$ for $i = 1, \ldots, N$, through a non-linear thersholding function $T_\lambda(\cdot)$. The dynamics of the internal state variables $u_i(t)$ is described by a set of non-linear ordinary differential equations (ODEs):

$$\begin{aligned} \tau\dot{\mathbf{u}}(t) &= \mathbf{\Phi}^T\mathbf{s} - \mathbf{u}(t) - (\mathbf{\Phi}^T\mathbf{\Phi} - \mathbf{I})\mathbf{a}(t) \\ \mathbf{a}(t) &= T_\lambda(\mathbf{u}(t)), \end{aligned} \tag{7}$$

where τ is the time constant. Through the evolution of (7), LCA networks efficiently solve the ℓ_1-norm regularized optimization problem [36] with much lower power compared to numerical iterative approaches [37].

Recent works of [9], [10] converted the LCA system to its spiking version and proposed the SSR system, which further improves the power efficiency by exploiting the advantages of SNN. In detail, SSR is designed to solve the SR problem (6) with the ℓ_1-norm penalty and the non-negative constraint to mimic the biological behaviors of spiking neurons [10], *i.e.*,

$$\min_{\mathbf{a}\geq 0} \frac{1}{2}\|\mathbf{s} - \mathbf{\Phi a}\|_2^2 + \lambda\|\mathbf{a}\|_1. \tag{8}$$

Exploiting rate-based IAF neurons, one can approximate the solution of (8) by the firing rate of neurons in the SSR system [10], where the main idea is to embed the gradient flow of (8) as the vector field of the network dynamics of SNN.

Following the methodology in Section II-A, we set the constant current $\mu^c = -\lambda$ with λ the same constant as in (8), and

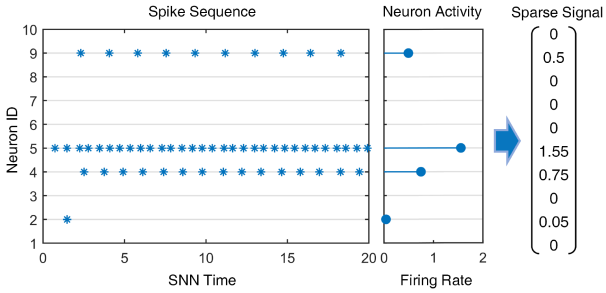


Fig. 3. Relationship between neuron activity and a sparse signal. The i-th element of the sparse signal is equal to the firing rate (12) of the i-th spiking neuron. SNN time t_{SNN} is related to the absolute time t_{ABS} by time constant τ, *i.e.*, $t_{SNN} = t_{ABS}/\tau$.

rewrite (4) as

$$\nu_i(t) = \int_0^t (\mu_i(s) - \lambda)ds - (\nu^s - \nu^r)\int_0^t \sigma_i(s)ds. \tag{9}$$

Then we define the input current $\mu_i(t)$ as

$$\mu_i(t) = b_i - \sum_{j\neq i} \Omega_{ij}\,(\alpha * \sigma_j)\,(t), \tag{10}$$

or in its differential form

$$\tau\dot{\mu}_i(t) = b_i - \mu_i(t) - \sum_{j\neq i} \Omega_{ij}\sigma_j(t). \tag{11}$$

Here $b_i \overset{\text{def}}{=} \boldsymbol{\phi}_i^T\mathbf{s}$ denotes the bias current, $\Omega_{ij} \overset{\text{def}}{=} \boldsymbol{\phi}_i^T\boldsymbol{\phi}_j$ is the synaptic weight that measures the influence of spikes according to the similarity of receptive fields, and $\alpha(t) \overset{\text{def}}{=} \frac{1}{\tau}e^{-t/\tau}\mathrm{H}(t)$ is a decay function where $\mathrm{H}(t)$ denotes the Heaviside step function. When no spike is fed, the input current $\mu_i(t)$ equals the bias current b_i, and if other neurons, *e.g.*, neuron-j, send a spike to neuron-i, the spike will be weighted by Ω_{ij} first, and then affect the input current $\mu_i(t)$ of neuron-i with an exponential decay.

Based on the above configuration, one can build an SSR system and obtain the output of network by the rate-based encoding method. That is, the i-th element of the sparse signal is equal to the firing rate $a_i(t)$ of neuron-i, *i.e.*,

$$a_i(t) = \frac{1}{t}\int_0^t \sigma_i(s)ds. \tag{12}$$

An example of neuron activity and a sparse signal is illustrated in Fig. 3. In the SSR system, the neuron activity $a_i(t)$ is strongly related to the membrane potential $\nu_i(t)$, where the potential charging speed is determined by the input current $\mu_i(t)$ and the constant current $-\lambda$. In fact, the differential form of the input current (11) can be regarded as another dynamic system similar to (7) but designed in a spiking-based manner, and the constant current $-\lambda$ serves as a thresholding function like $T_\lambda(\cdot)$ to induce sparsity. Thus, SSR can solve the SR problem (8) via imitating the basic mechanism of LCA (7) in SNN. A formal and rigorous analysis of the equivalence between the outputs of

Authorized licensed use limited to: Weizmann Institute of Science. Downloaded on January 22,2023 at 10:47:14 UTC from IEEE Xplore. Restrictions apply.

SSR and LCA is provided in [10]. Here we summarize the result as the following theorem.

Theorem 1: Consider an SSR system defined by (2), (3), (9), (10) and (12). As time goes to infinity, the firing rate $\mathbf{a}(t) = [a_1(t), \ldots, a_N(t)]^T$ of SSR is equivalent to the output of LCA and converges to the optimal solution of the SR problem (8).

Proof: See [10] for details. □

Compared to the analog LCA system (7), SSR poses significant advantages in power efficiency. Take the system output as an example. The LCA system needs to constantly keep a relatively high voltage/current to carry the output signal, while the SSR system emits sparse and binary spikes to represent the output, which is easy to implement in a low-power manner. In [10], the SSR algorithm is shown to largely outperform the analog LCA with a 94% decrease in power consumption and an 89% decrease in convergence time, showing the great potential of the spiking-based framework. To tackle the underestimation issue of ℓ_1-norm regularization, we next extend SSR to solve the SR problem with non-convex penalties.

III. SSR With Non-Convex Penalties

Consider the SR problem with non-convex penalty $g(\cdot)$, *i.e.*,

$$\min_{\mathbf{a}\geq 0} E(\mathbf{a}) = \frac{1}{2}\|\mathbf{s} - \mathbf{\Phi a}\|_2^2 + \lambda \sum_{i=1}^{N} g(|a_i|). \tag{13}$$

One can optimize (13) via IRLS [16] or IRL1 [14] and find more accurate estimates than optimizing (8). Thus applying spiking neurons to solve (13) can further improve the performance of SSR. To this end, we first extend SSR to its adaptive version (*i.e.*, A-SSR) by adapting the spiking neuron, and then establish the bridge between A-SSR and the SR problem with non-convex penalties (13).

A. Adaptive Spiking Sparse Recovery

The configuration of the input current $\mu_i(t)$ and the encoding method in A-SSR remain the same as in SSR, *i.e.*, (10) and (12). To fit the non-convex regularization in (13), we need to extend (9) by replacing the the constant current $-\lambda$ with the *adaptive current* $-\Lambda_i(t)$, *i.e.*,

$$\nu_i(t) = \int_0^t (\mu_i(s) - \Lambda_i(t))ds - (\nu^s - \nu^r)\int_0^t \sigma_i(s)ds, \tag{14}$$

where $\Lambda_i(t) = \lambda g'(|a_i(t)|)$ with $g'(|a_i(t)|)$ the first derivative of $g(|a_i(t)|)$ w.r.t. $|a_i(t)|$ and λ the same constant as in (13). Equation (14) indicates that the charging speed of potential $\nu_i(t)$ is also affected by the firing rate $a_i(t)$, and the adaptive function is determined by the chosen penalty $g(\cdot)$. In other words, the spiking neuron in A-SSR can adaptively adjust the charging speed according to its own firing activity.

To obtain the adaptive current $\Lambda_i(t)$ in analog SNNs, we need to compute the non-linear function $g'(\cdot)$. These non-linear functions can be easily realized even by using simple operational amplifiers in analog systems, with which the basic functionalities such as `multiplication, division, log, exp, abs` already exist. For instance, the first derivative of the exponential penalty (as Example 1 in Section III-B), *i.e.*, $g'(x) = \gamma e^{-\gamma x}$, can be realized via cascading an exponential operator and several multiplication operators [38].

For simplicity, we set the spiking threshold $\nu^s = 1$, the resting potential $\nu^r = 0$ and additionally add the lower bound of potential $\nu^- = 0$ so that $\nu_i(t)$ will be reset to ν^- once $\nu_i(t) < \nu^-$. Based on the above, we summarize the definition of A-SSR below.

$$\textbf{Input current} \quad \mu_i(t) = b_i - \sum_{j\neq i} \Omega_{ij}\,(\alpha * \sigma_j)\,(t) \tag{15a}$$

$$\textbf{Potential} \quad \nu_i(t) = \int_0^t (\mu_i(s) - \Lambda_i(t))ds - \int_0^t \sigma_i(s)ds \tag{15b}$$

$$\textbf{Firing time} \quad t_{i,k} : \nu_i(t_{i,k}) = \nu^s \tag{15c}$$

$$\textbf{Spike sequence} \quad \sigma_i(t) = \sum_k \delta(t - t_{i,k}) \tag{15d}$$

$$\textbf{Firing rate} \quad a_i(t) = \frac{1}{t}\int_0^t \sigma_i(s)ds. \tag{15e}$$

Note that the term $(\nu^s - \nu^r)$ in (15b) disappears because $\nu^s = 1$ and $\nu^r = 0$ in our configuration. Through the dynamic evolution of A-SSR systems, the solution of the SR problem (13) can be approximated by the firing rate (15e) of spiking neurons (see Theorem 2 in Section III-C).

Apart from the proposed A-SSR system, there exist several possible implementations of the spiking SR system with non-convex penalties. For instance, one could apply adaptation on the spiking threshold ν^s instead of the adaptive current used in A-SSR, or modify the SSR system [9] based on the nonlinearities corresponding to non-convex objectives in [21]. Here, we focus on the implementation of A-SSR and investigate its theoretical properties including convergence conditions (Section III-B), system stability, and solution equivalence (Section III-C).

B. Choice of Non-Convex Penalties

Instead of ℓ_1-norm regularization, the target optimization (13) is regularized with non-convex penalties $g(\cdot)$, and thus the system stability is directly related to the chosen $g(\cdot)$. To guide the penalty selection, we provide the following rules.

- **Rule-1** $g(\cdot)$ is non-negative and subanalytic on $[0, +\infty)$.
- **Rule-2** The first derivative of $g(\cdot)$ is continuous and non-negative on $[0, +\infty)$, *i.e.*, $g'(\cdot) \geq 0$.
- **Rule-3** The second derivative of $g(\cdot)$ exists, and is bounded $g''(x) \in (-\frac{1}{\lambda}, 0), \forall x \in (0, +\infty)$.

With Rule-1, the objective (13) is also subanalytic, guaranteeing the convergence of A-SSR (see Appendix A for details). Rule-2 ensures that $\Lambda_i(t)$ will not go negative and prevents the potential $\nu_i(t)$ from overcharging. Finally, Rule-3 suggests that the penalty choice should be related to parameter λ, which is beneficial to the system stability. Rule-3 also implies that $g'(|a_i|)$ is monotonically decreasing over $|a_i|$ in our case. Therefore, these neurons with higher firing rates are more likely to fire spikes

Authorized licensed use limited to: Weizmann Institute of Science. Downloaded on January 22,2023 at 10:47:14 UTC from IEEE Xplore. Restrictions apply.

according to (15b), leading to better recovery of high-amplitude components.

Although Rule-3 may seem strict, there is a wide range of possible penalties under an appropriate λ, including those commonly used in practice [21], [39]. Here, we analyze three exemplary penalties that will be used in our experiments.

Example 1: The exponential penalty $g(x) = 1 - e^{-\gamma x}$, where $\gamma > 0$ is a constant. It is obvious that the exponential penalty meets Rule-1 and Rule-2. For Rule-3, note that

$$g''(x) = -\gamma^2 e^{-\gamma x} \in (-\gamma^2, 0), \quad \forall x \in (0, +\infty).$$

Given the parameter λ, the exponential penalty satisfies all the rules when $\gamma \in \left(0, \frac{\sqrt{\lambda}}{\lambda}\right]$.

Example 2: The logarithmic penalty $g(x) = \log(x + \theta)$, where $\theta > 0$ is a constant. To meet Rule-1, we must have $g(0) = \log(\theta) \geq 0$, therefore $\theta \geq 1$. With $\theta \geq 1$, Rule-2 holds as well. As for Rule-3, we have

$$g''(x) = -\frac{1}{(x+\theta)^2} \in \left(-\frac{1}{\theta^2}, 0\right), \quad \forall x \in (0, +\infty).$$

Given the parameter λ, one can derive that $\theta \geq \sqrt{\lambda}$ satisfying Rule-3. Hence, the logarithmic penalty satisfies all the rules when $\theta \in [\xi, +\infty)$ with $\xi = \max(\sqrt{\lambda}, 1)$.

Example 3: The arctangent penalty $g(x) = \arctan(x/\eta)$, where $\eta > 0$ is a constant. It is clear that this penalty satisfies Rule-1 and Rule-2. To meet Rule-3, note that

$$g''(x) = -\frac{2\eta^2 x}{\left(\eta^2 + x^2\right)^2} \in \left[-\frac{3\sqrt{3}}{8\eta}, 0\right), \quad \forall x \in (0, +\infty).$$

For a given λ, the arctangent penalty will meet all the rules when $\eta \in \left(\frac{3\sqrt{3}}{8}\lambda, +\infty\right)$.

With the above examples, one can see how the rules guide the selection of non-convex penalties. Under the same penalty $g(\cdot)$ that satisfies all the above rules, we next discuss the equivalence between the output of A-SSR and the solution of the SR problem (13), and analyze the convergence property of A-SSR systems.

C. Convergence Analysis of A-SSR

In this section, we prove the global asymptotic convergence of A-SSR systems and state that the firing rate $\mathbf{a}(t)$ of A-SSR is equivalent to a critical point of the SR problem (13) as time goes to infinity. Let us first review the definition of critical points: A vector $\mathbf{a}^* \in \mathbb{R}^N$ is called as a critical point of objective $V(\cdot)$ when it satisfies the inclusion

$$0 \in \partial V(\mathbf{a}^*).$$

We then state the convergence result as follows.

Theorem 2: Consider a penalty $g(\cdot)$ satisfying all the rules mentioned in Section III-B. We additionally define the average input current $u_i(t)$ as

$$u_i(t) \stackrel{\text{def}}{=} \frac{1}{t}\int_0^t \mu_i(s)ds. \tag{16}$$

Then the average input current $\mathbf{u}(t) = [u_1(t), \ldots, u_N(t)]^T$ and the firing rate $\mathbf{a}(t) = [a_1(t), \ldots, a_N(t)]^T$ of A-SSR are globally asymptotically convergent, and $\mathbf{a}(t)$ will converge to a critical point of (13) as time goes to infinity.

Proof: See Appendix A for the detailed proof. □

The contribution of Theorem 2 is two-fold:

- *System Stability:* The stability of A-SSR is guaranteed. Though we cannot promise to find the global minimum of $E(\mathbf{a})$ defined in (13), which is generally hard (or even impossible) due to the non-convexity of $E(\mathbf{a})$, the convergence result is often sufficient for practical purposes. In our experiments, A-SSR systems can obtain a satisfactory solution within 20 SNN time (see Fig. 7(c)), which is equivalent to around 0.04 s on the neuromorphic chip Loihi (see Fig. 8).
- *Solution Equivalence:* The output of A-SSR is equivalent to the solution of the SR problem (13), *e.g.*, see Fig. 6. By using spiking neurons to solve the non-convex regularized SR problem, A-SSR can take advantage of both fields and serve as an efficient SR solver with high recovery accuracy (see Fig. 7) and low power consumption (see Table I).

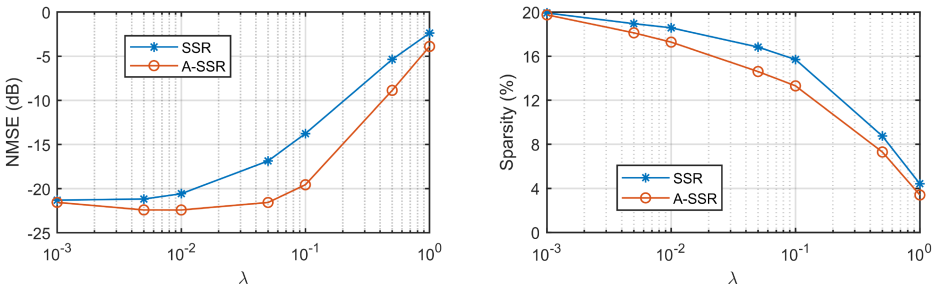
IV. Experimental Analysis

To verify the performance of A-SSR, we present a series of experiments based on Nengo [40], a neural engineering simulation platform, and the neuromorphic processor Loihi [22]. Our experiments are conducted on both synthetic and real datasets. In the experiment of synthetic data, we first investigate the influence of the trade-off parameter λ in (13), which is related to the accuracy and sparsity of the final estimates. The different working process between SSR and A-SSR is then analyzed to reveal the benefit of the adaptive mechanism in (15b). Following that, we conduct experiments to verify the solution equivalence between A-SSR and the SR problem (13). Next, we compare the performance of A-SSR systems corresponding to different penalty functions, and show the advantages of A-SSR in power consumption and running time. Finally, we solve an image reconstruction task and a face recognition task to demonstrate the potential of A-SSR in practical applications. In our experiments, the time constant τ is set to 0.1; the entries of dictionary $\mathbf{\Phi}$ are sampled randomly from a standard Gaussian distribution $\mathcal{N}(0, 1)$; and the noise we consider here is the white Gaussian noise.

A. Influence of Parameter λ

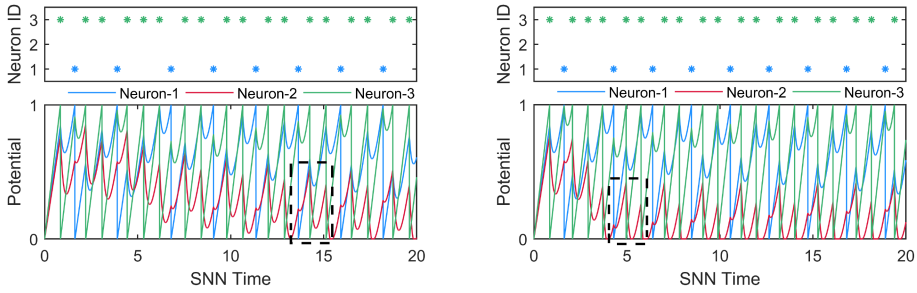
In this experiment, we mainly illustrate the relationship between parameter λ and the performance of A-SSR. We design an A-SSR with exponential penalty $g(|a_i|) = 1 - e^{-\gamma|a_i|}$ where $\gamma = 1$, and compare to SSR under the same dictionary $\mathbf{\Phi} \in \mathbb{R}^{100\times 200}$. From Fig. 4, it is obvious that the A-SSR system is superior to SSR in both accuracy and sparsity. The result also shows that the performance is closely related to the parameter λ. If λ is chosen too large, the charging speed of all neurons becomes slow according to (15b), resulting in longer convergence time. In contrast, if λ is too small, the effect of the adaptive mechanism becomes negligible since the sparsity constraint in (13) almost vanishes. The selection of λ also affects the choice of non-convex penalty due to Rule-3 in Section III-B. Generally

Authorized licensed use limited to: Weizmann Institute of Science. Downloaded on January 22,2023 at 10:47:14 UTC from IEEE Xplore. Restrictions apply.



(a) Accuracy (b) Sparsity

Fig. 4. Influence of parameter λ. (a) Recovery accuracy under different λ. The metric $NMSE = 10\log_{10}(\|\hat{\mathbf{a}} - \mathbf{a}\|_2^2/\|\hat{\mathbf{a}}\|_2^2)$ indicates normalized mean square error, where $\hat{\mathbf{a}}$ denotes the original signal. (b) Comparison of recovery sparsity, where sparsity is defined as the percentage of non-zero coefficients in total.



(a) SSR (b) A-SSR

Fig. 5. The potential diagram (bottom) and the spike raster diagram (top) of SSR and A-SSR with exponential penalty. The dashed box indicates the first stable period.

smaller λ leads to a wider choice of $g(\cdot)$. Taking all factors into consideration, we choose $\lambda = 0.1$ in the following experiments.

B. Comparison Between SSR and A-SSR

Using the same A-SSR system as in Section IV-A, we now compare the working process of the proposed A-SSR to SSR. For simplicity, we design a simple SNN with only three neurons, and set the original signal to $\hat{\mathbf{a}} = [0.4792, 0, 0.9754]^T$. As displayed in Fig. 5(a) and (b), one can find that these active neurons (*e.g.*, neuron-1) in A-SSR tend to produce more spikes than that of the SSR system. This is because the adaptive mechanism within A-SSR encourages these active neurons to spike more frequently, leading to better recovery of high-amplitude components. On the other hand, these inactive neurons (*e.g.*, neuron-2) are more intensively affected as input spikes increase, making A-SSR be in periodical stabilization earlier (as shown in Fig. 5). After the same SNN time, the normalized mean square errors ($NMSEs$) generated by SSR and A-SSR are -19.95 dB and -28.97 dB, correspondingly.

C. Verification of Solution Equivalence

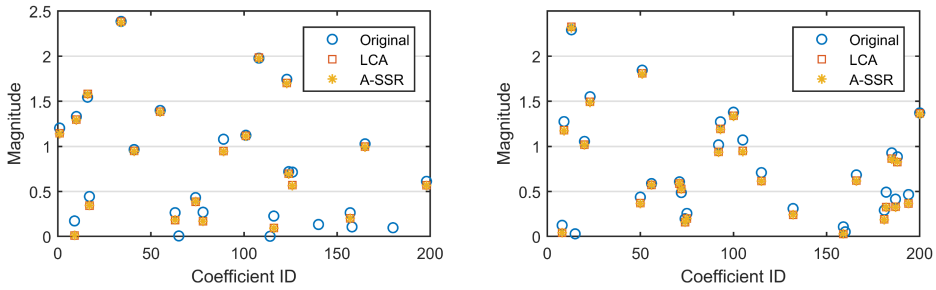
To verify the solution equivalence, we implement LCA [20], [21] on Matlab as a baseline system to solve the SR problem (13). Apart from the A-SSR system in Section IV-A, we employ an additional A-SSR system with the logarithmic penalty $g(|a_i|) = \log(|a_i| + 1)$, which belongs to the family of approximate ℓ_p-norm ($0 \leq q \leq 1$) functions in [21]. We configure A-SSR and LCA to solve the same non-convex regularized optimization problems (13) under the same dictionary $\mathbf{\Phi} \in \mathbb{R}^{100\times 200}$, 15% sparsity, and noise-free scenarios. As illustrated in Fig. 6, A-SSR and LCA converge to similar solutions, indicating that the output of A-SSR is equivalent to the solution of SR problem (13).

D. Performance of A-SSR Under Different Adaptive Functions

To explore the performance of A-SSR with different adaptive functions, we add A-SSR with $g(|a_i|) = \arctan(|a_i|/\eta)$ into the experiments and set $\eta = 1$. We first compare the probability of successful recovery with the same dictionary $\mathbf{\Phi} \in \mathbb{R}^{100\times 200}$ and different sparsity levels. Here we define successful recovery as the case $NMSE < -15$ dB in the noise-free scenario, and test each algorithm with 100 samples. In Fig. 7(b), the recovery probability of SSR drops quickly at the beginning while all A-SSR systems hold their probability at nearly 1 until the sparsity is over 15%, showing that A-SSR systems can still recover the signal reliably under less sparse conditions.

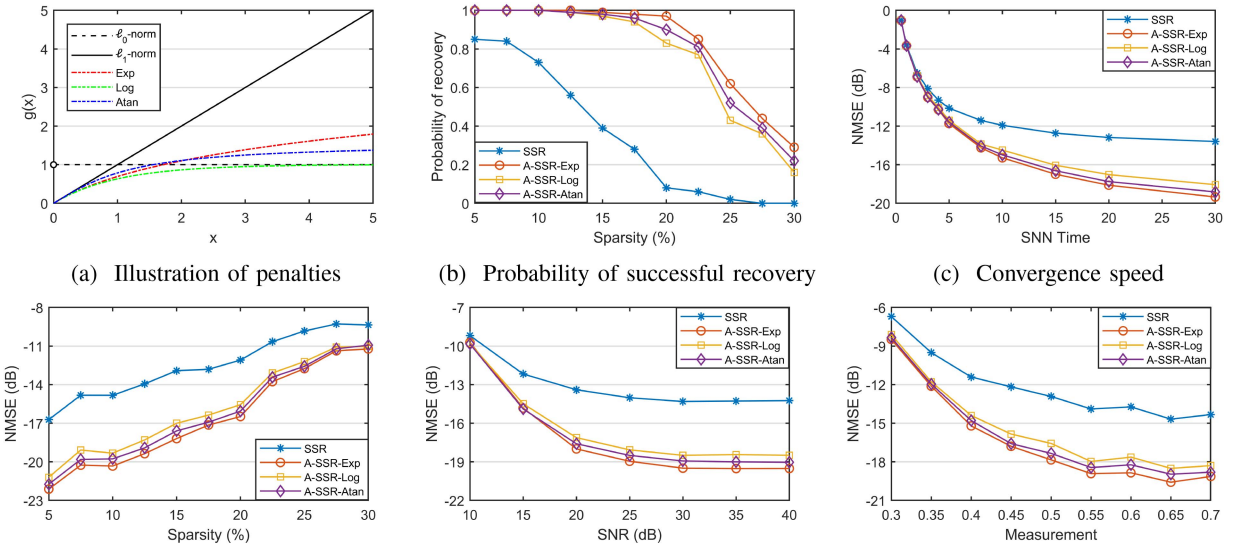
Next, we consider the situation where the signal is corrupted by white Gaussian noise. We choose $\mathbf{\Phi} \in \mathbb{R}^{100\times 200}$, 20 dB SNR and 15% sparsity as the default setting, and present four experiments: Fig. 7(c) shows the convergence comparison under the same conditions, and Fig. 7(d)–(f) show the performance under different levels of sparsity, noise, and measurements.

Authorized licensed use limited to: Weizmann Institute of Science. Downloaded on January 22,2023 at 10:47:14 UTC from IEEE Xplore. Restrictions apply.



(a) Exponential penalty (b) Logarithmic penalty

Fig. 6. Convergence results of LCA and A-SSR. Both A-SSR and LCA are configured to solve the same optimization problems regularized by (a) exponential and (b) logarithmic penalties. Zero-valued elements are hidden for clarity.



(a) Illustration of penalties (b) Probability of successful recovery (c) Convergence speed

(d) Result under different sparsity (e) Result under different noise (f) Result under different measurement

Fig. 7. Performance of A-SSR systems with different adaptive functions. (a) Illustration of the penalties used in this experiment; ℓ_0-norm and ℓ_1-norm are also drawn for reference. (b) Probability of successful recovery in the noise-free scenario. (c) Convergence comparison. (d)–(f) Performance under different levels of sparsity, noise and measurement, respectively. SNR indicates the signal-to-noise ratio, and measurement represents the size ratio M/N of dictionary $\mathbf{\Phi} \in \mathbb{R}^{M\times N}$.

Evidently, all the A-SSR systems achieve general improvement in accuracy compared to SSR. The performance of A-SSR also varies with the chosen non-convex penalty, *i.e.*, the adaptive function, and the exponential penalty is the best choice for the proposed experiments.

E. Computational Efficiency of A-SSR

In this section, we conduct experiments to verify the computational efficiency of the proposed A-SSR system compared to other spiking-based and iterative-based SR approaches. Although the existing LCA [21] also provides an analog framework for SR and can be extended to solve non-convex regularized SR problems, the prior work of [9] has converted LCA to a spiking-based version (*i.e.*, SSR) and verified the significant superiority of SSR over analog LCA in both power consumption and convergence speed. Therefore, we mainly compare our A-SSR with SSR [9], [10] and FISTA [23], which is one of the fastest numerical solvers in the SR field. To show the advantages of A-SSR in power consumption and running time, we implement SSR and A-SSR with exponential penalty on the Loihi chip, and use an Intel Core i7-6700HQ 2.6 GHz CPU to run the reference iterative algorithm FISTA [23]. Since FISTA and SSR are designed to solve the ℓ_1-norm regularized optimization problem while A-SSR is designed for the non-convex one, the following strategy is used to make the comparison fair: we first run the SSR system and FISTA to obtain the solutions satisfying $NMSE < -13$ dB under 15% sparsity and the same dictionary with 0.5 measurement ratio. Then we record the SNN time needed for SSR to meet the criterion, and run A-SSR for the same SNN time. In this way, we can measure the power consumption of all algorithms. From the result of Table I, it is evident that spiking approaches are energy efficient with around two orders of magnitude lower power compared to FISTA.

One more interesting point is that although A-SSR systems need to operate an additional adaptive mechanism, its power

Authorized licensed use limited to: Weizmann Institute of Science. Downloaded on January 22,2023 at 10:47:14 UTC from IEEE Xplore. Restrictions apply.

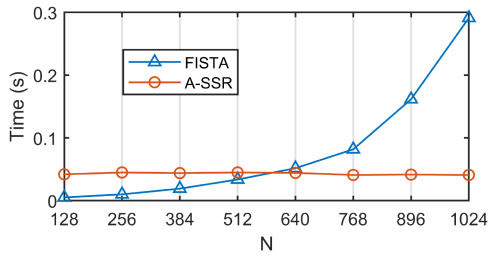


Fig. 8. Comparison of running time between FISTA [23] and A-SSR under different sizes of dictionary $\mathbf{\Phi} \in \mathbb{R}^{M\times N}$.

TABLE II
QUANTITATIVE RESULT OF RECONSTRUCTION

Metric	PSNR(dB)	SSIM	NMSE(dB)
SSR [10]	18.87	0.7722	-9.039
A-SSR-Exp	**24.49**	**0.9196**	**-14.65**
A-SSR-Log	22.87	0.9014	-13.04
A-SSR-Atan	23.68	0.9106	-13.84

The above results are acquired in the noise-free scenario. The highest PSNR and SSIM, and the lowest NMSE are marked in bold.

consumption is still comparable to (or even lower than) that of SSR. To explain this, we recall the comparison of SSR and A-SSR in Fig. 5. With the adaptive mechanism, A-SSR system is likely to be stable earlier than SSR. During the stable period, these inactive neurons (*e.g.*, neuron-2) only operate within a low potential range or even remain at $\nu^- = 0$. This behavior saves energy to some extent since the membrane potential is related to voltage on the Loihi chip. In addition, the A-SSR system actually produces results with around -17 dB NMSE since we compare SSR and A-SSR under the same SNN time. Therefore, our A-SSR algorithm is able to achieve more accurate estimates than SSR and maintain power-efficient implementation on hardware.

We also measure the running time of FISTA and A-SSR in the above power experiments. As displayed in Fig. 8, the running time of FISTA increases significantly with the rising scale of the SR problem, while that of A-SSR stays at around 0.04 s at different scales. This is because Loihi assigns more resource cores to meet the increasing size of neurons and runs all resource cores in a highly parallel manner. Thus the running time of A-SSR is related to the number of neurons in one resource core and the spike traffic rather than the scale of the SR problem, showing great potential in practical applications.

F. Application: Image Reconstruction

SR plays an important role in image reconstruction [42]. To evaluate A-SSR in practical tasks, we perform image reconstruction on the MNIST database [41], which consists of 28×28 gray images with handwritten digits from different writers. For simplicity, we treat each image as a 784-dimensional vector and construct the dictionary with 0.5 measurement ratio, *i.e.*, $\mathbf{\Phi} \in \mathbb{R}^{392\times 784}$. With a test image $\mathbf{a}^* \in \mathbb{R}^{784}$, the input vector $\mathbf{s} \in \mathbb{R}^{392}$ can be obtained by $\mathbf{s} = \mathbf{\Phi a}^*$. Under the same SNN time, we test SSR and A-SSR algorithms with 100 samples per digit.

Some reconstruction results are shown in Fig. 9. Compared to the SSR algorithm, A-SSR systems are able to reconstruct images with higher contrast, more details, and less noises since A-SSR systems can produce more accurate and sparse estimates. We then employ the commonly used metrics peak signal-to-noise ratio (PSNR) and structural similarity (SSIM) [43] for quantitative evaluation. In Table II, all the A-SSR methods surpass SSR over 4 dB in PSNR and 0.12 in SSIM. It demonstrates that the adaptive mechanism not only achieves better recovery of high-amplitude elements, but also maintains the overall structure of images. In Fig. 10, we consider the case where the test image $\mathbf{a}^*$ is corrupted by white Gaussian noise. Though the reconstruction performance fluctuates with the noise level, A-SSR systems still achieve an overall improvement in reconstruction quality, which is consistent with the analysis in Section IV-D.

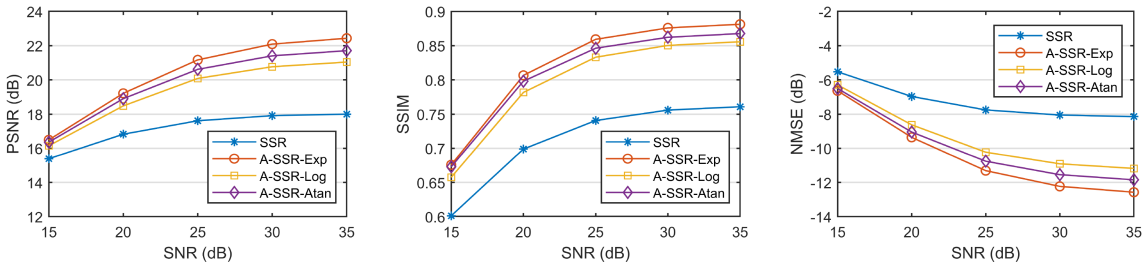
G. Application: Face Recognition

In addition to image reconstructions, SR shows advantages in image recognition [44]. In our recognition task, we select 640 frontal-face images of 10 individuals from the Extended Yale B database [45]. All images are downsampled to shape 18×16 and regarded as 288-dimensional vectors, and we denote the m-th face image of the n-th subject as $\mathbf{A}_{m,n} \in \mathbb{R}^{288}$. For each individual, 40 images are taken for training and the rest (*i.e.*, 24 images) are left for the testing phase. In the training, we first normalize each $\mathbf{A}_{m,n}$ to a unit Euclidean norm $\hat{\mathbf{A}}_{m,n} \in \mathbb{R}^{288}$, and then construct the dictionary by concatenating all the training samples, *i.e.*, $\mathbf{\Phi} = [\hat{\mathbf{A}}_{1,1}, \hat{\mathbf{A}}_{2,1}, \ldots, \hat{\mathbf{A}}_{40,1}, \hat{\mathbf{A}}_{1,2}, \ldots, \hat{\mathbf{A}}_{40,10}] \in \mathbb{R}^{288\times 400}$. For the testing, we take the remaining normalized images, *i.e.*, $\hat{\mathbf{A}}_{m,n}$ where $m = 41, 42, \ldots, 64$ and $n = 1, 2, \ldots, 10$, as $\mathbf{s}$ to form the bias current $\mathbf{b} = \mathbf{\Phi}^T\mathbf{s}$ in our A-SSR system. After running 30 SNN time, we evaluate the recognition performance by analyzing the output spike sequences. Since the recognition rate varies with the chosen decision strategy, we instead analyze the confidence of recognition, which is a more intuitive representation of the performance. Here we define the recognition confidence as the ratio of spikes emitted by a certain class to the total number of output spikes.

Fig. 11 illustrates the recognition confidence between the SSR and our A-SSR systems in the noise-free scenario. With the help of the adaptive mechanism, all A-SSR algorithms achieve around 5% confidence improvement over SSR, leading to stronger system robustness. To further verify this, we add two kinds of interference to the original test image. In the first case, we corrupt the test image with white Gaussian noise according to a pre-setting signal-to-noise ratio (SNR) value (as shown in Fig. 12(a)). In the second scenario, we randomly choose a percentage of pixels from the test image, and replace their values with zeros (as shown in Fig. 12(b)). Thus we can simulate the situation where input information is limited. From the results of Fig. 12(c) and (d), one can see that A-SSR methods maintain their advantages either under noises or with limited information.

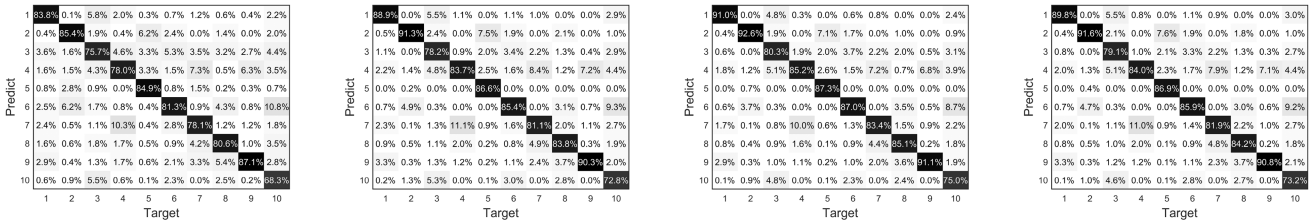
Authorized licensed use limited to: Weizmann Institute of Science. Downloaded on January 22,2023 at 10:47:14 UTC from IEEE Xplore. Restrictions apply.

Fig. 9. Qualitative comparisons between SSR (row 2) and A-SSR (row 3-5 are the results of A-SSR-Exp, A-SSR-Log and A-SSR-Atan respectively) on the MNIST database [41]. Row 1 displays the ground truth.



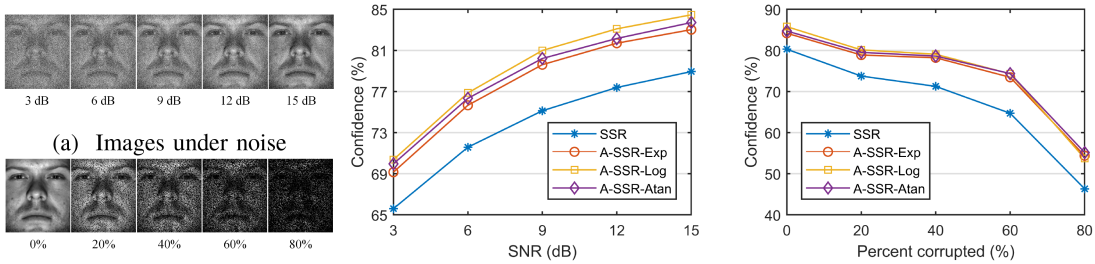
(a) PSNR results (b) SSIM results (c) NMSE results

Fig. 10. Quantitative comparisons between SSR and A-SSR under different levels of noise.



(a) SSR (b) A-SSR-Exp (c) A-SSR-Log (d) A-SSR-Atan

Fig. 11. Confidence matrices of SSR, A-SSR-Exp, A-SSR-Log and A-SSR-Atan in the noise-free scenario.



(a) Images under noise

(b) Images under pixel corruption (c) Confidence under noise (d) Confidence under pixel corruption

Fig. 12. Recognition under corruption. (a), (b) show test images under different levels of SNR and pixel corruption. (c), (d) display the mean confidence of correct recognition under varying SNR and percentage of pixel corruption.

Authorized licensed use limited to: Weizmann Institute of Science. Downloaded on January 22,2023 at 10:47:14 UTC from IEEE Xplore. Restrictions apply.

V. Conclusion

In this work, we propose A-SSR to solve a class of non-convex regularized SR problems via adaptive spiking neurons. By establishing the connection between the non-convex penalty and the adaptive mechanism, A-SSR can avoid the underestimation of high-amplitude components in sparse signals, thereby obtaining results with better accuracy and sparsity. Exploiting the advantage of spiking neurons, A-SSR implemented on Loihi only consumes around one percent of the power of FISTA. Thus the proposed method is able to produce better estimates and remain energy-efficient on hardware. To guide the use of A-SSR, we also provide a set of selection rules for non-convex penalties and prove the global asymptotic convergence of A-SSR. Extensive experiments demonstrate the remarkable performance and low power consumption of A-SSR, showing potential in practical usages like always-on or energy-constrained applications.

Appendix A
Proof of Theorem 2

To help prove Theorem 2, we design a non-spiking auxiliary system which serves as a bridge between A-SSR and the non-convex regularized optimization (13). The auxiliary system is defined as follows (dependence on t is omitted for readability):

$$\tau \dot{u}_i^{au} = b_i - u_i^{au} - \sum_{j\neq i} \Omega_{ij} a_j^{au},$$
$$a_i^{au} = \max(u_i^{au} - \Lambda_i^{au}, 0) = \begin{cases} f\left(u_i^{au}\right), & \text{if } u_i^{au} > \Lambda_i^{au}, \\ 0, & \text{else,} \end{cases} \tag{17}$$

where $f(u_i^{au}) = u_i^{au} - \Lambda_i^{au} = u_i^{au} - \lambda g'(|a_i^{au}|)$ is defined on $[0, +\infty)$, and variables are restricted to be non-negative, *i.e.*, $u_i^{au} \geq 0$ and $a_i^{au} \geq 0$.

For clarity, we divide the proof into four lemmas. The relationship between the auxiliary system and the target optimization (13) is first established in Lemma 1. In Lemma 2, we then prove the global asymptotic convergence of the auxiliary system by leveraging the convergence result in [36]. With the boundedness of $\boldsymbol{\mu}(t)$ and $\mathbf{u}(t)$ in Lemma 3, we finally prove that the A-SSR system converges to the auxiliary system as time goes to infinity, and the firing rate $\mathbf{a}(t)$ will converge to a critical point of the corresponding non-convex regularized optimization (13).

Lemma 1: With the objective $E(\mathbf{a}^{au})$ defined in (13), the auxiliary system (17) satisfies that $\tau\dot{\mathbf{u}}^{au} \in -\partial E(\mathbf{a}^{au})$.

Proof: First define the following operator:

$$\operatorname{sgn}(x) \begin{cases} = 1 & \text{if } x > 0, \\ \in [-1, 1] & \text{if } x = 0, \\ = -1 & \text{if } x < 0. \end{cases} \tag{18}$$

When $\boldsymbol{x}$ is a vector, $\operatorname{sgn}(\boldsymbol{x})$ is also a vector with the i-th element equal to $\operatorname{sgn}(x_i)$. Then we can derive the inclusion

$$u_i^{au}(t) - a_i^{au}(t) \in \Lambda_i^{au}(t) \operatorname{sgn}(a_i^{au}(t)) \tag{19}$$

based on (17) and the following discussion:

- When $u_i^{au}(t) > \Lambda_i^{au}(t)$, $\operatorname{sgn}(a_i^{au}(t)) = \operatorname{sgn}(u_i^{au}(t)) = 1$ holds since $\Lambda_i^{au}(t) \geq 0$ from Rule-2 in Section III-B. Hence, we can derive the relationship $u_i^{au}(t) - a_i^{au}(t) = \Lambda_i^{au}(t) = \Lambda_i^{au}(t) \operatorname{sgn}(a_i^{au}(t))$.
- When $u_i^{au}(t) \in [0, \Lambda_i^{au}(t)]$, we have $a_i^{au}(t) = 0$, therefore $u_i^{au}(t) - a_i^{au}(t) = u_i^{au}(t) \in \Lambda_i^{au}(t) \operatorname{sgn}(a_i^{au}(t))$.

According to (19), the state $u_i^{au}(t)$ in (17) can be rewritten as

$$\begin{aligned} \tau\dot{\mathbf{u}}^{au} &= \mathbf{b} - \mathbf{u}^{au} - (\boldsymbol{\Phi}^T\boldsymbol{\Phi} - \boldsymbol{I})\mathbf{a}^{au} \\ &\in \mathbf{b} - \boldsymbol{\Phi}^T\boldsymbol{\Phi}\mathbf{a}^{au} - \boldsymbol{\Lambda}^{au} \operatorname{sgn}(\mathbf{a}^{au}), \end{aligned}$$

which is also equivalent to

$$\tau\dot{\mathbf{u}}^{au} \in \boldsymbol{\Phi}^T\mathbf{s} - \boldsymbol{\Phi}^T\boldsymbol{\Phi}\mathbf{a}^{au} - \lambda g'(|\mathbf{a}^{au}|) \operatorname{sgn}(\mathbf{a}^{au})$$

since $\mathbf{b} = \boldsymbol{\Phi}^T\mathbf{s}$ and $\boldsymbol{\Lambda}^{au} = \lambda g'(|\mathbf{a}^{au}|)$. On the other hand, the sub-differentiation of $E(\mathbf{a}^{au})$ w.r.t. $\mathbf{a}^{au}$ yields

$$\partial E(\mathbf{a}^{au}) = \left(-\boldsymbol{\Phi}^T\mathbf{s} + \boldsymbol{\Phi}^T\boldsymbol{\Phi}\mathbf{a}^{au} + \lambda g'(|\mathbf{a}^{au}|) \operatorname{sgn}(\mathbf{a}^{au})\right)^T.$$

It is thus straightforward that $\tau\dot{\mathbf{u}}^{au} \in -\partial E(\mathbf{a}^{au})$ holds. □

We next prove the convergence property of the auxiliary system. In fact, the auxiliary system is a one-side case of LCA [20], [46]. The global asymptotic convergence of LCA has been proven in [36] using Łojasiewicz inequality. Here we transform Theorem 1 and 2 in [36] to the one-side case that we are focusing on.

Theorem 3: (LCA convergence results from [36]) Consider the LCA system (17). If it satisfies the following conditions:

1) $f(\cdot)$ is continuous on $[0, +\infty)$ and differentiable on $(0, +\infty)$,
2) $f(\Lambda_i^{au}(t)) = 0$, and $f(u_i^{au}(t)) \leq u_i^{au}(t)$, $\forall u_i^{au}(t) \in [\Lambda_i^{au}(t), +\infty)$,
3) $u_i^{au}(t) - a_i^{au}(t) \in \lambda\partial C(a_i^{au}(t))$,
4) $f(\cdot)$ is subanalytic on $[0, +\infty)$,
5) $f'(u_i^{au}(t)) > 0$, $\forall u_i^{au}(t) \in (\Lambda_i^{au}(t), +\infty)$,
6) $\forall U > \Lambda_i^{au}(t)$, $\exists \zeta_U > 0$ *s.t.* $\forall u_i^{au}(t) \in [\Lambda_i^{au}(t), U]$, $f'(u_i^{au}(t)) \leq \zeta_U$,

then the state $\mathbf{u}^{au}(t) = [u_1^{au}(t), \ldots, u_N^{au}(t)]^T$ and the output $\mathbf{a}^{au}(t) = [a_1^{au}(t), \ldots, a_N^{au}(t)]^T$ of system (17) are globally asymptotically convergent.

Now we show the convergence of auxiliary system (17) by proving the conditions required in Theorem 3.

Lemma 2: The state $\mathbf{u}^{au}(t)$ and the output $\mathbf{a}^{au}(t)$ of the auxiliary system (17) are globally asymptotically convergent.

Proof: With the rules in Section III-B, it is straightforward that the system (17) satisfies conditions 1)-2). Condition 3) is met by (19) since $C(a_i^{au}(t)) = g(|a_i^{au}(t)|)$ in our case. As for condition 4), it is used to guarantee that $E(\mathbf{a})$ is subanalytic in [36], which is also satisfied by Rule-1 in Section III-B.

To prove condition 5), we derive the first derivative of $f(u_i^{au}(t))$ on $u_i^{au}(t) \in (\Lambda_i^{au}(t), +\infty)$:

$$\begin{aligned} f'(u_i^{au}(t)) &= 1 - \frac{\mathrm{d}\Lambda_i(t)}{\mathrm{d}a_i^{au}(t)} \frac{\mathrm{d}a_i^{au}(t)}{\mathrm{d}u_i^{au}(t)} \\ &= 1 - \lambda g''(|a_i^{au}(t)|) \operatorname{sgn}(a_i^{au}(t)) f'(u_i^{au}(t)). \end{aligned}$$

With Rule-3 in Section III-B, one can obtain that

$$f'(u_i^{au}(t)) = \frac{1}{1 + \lambda g''(|a_i^{au}(t)|) \operatorname{sgn}(a_i^{au}(t))} > 0,$$

Authorized licensed use limited to: Weizmann Institute of Science. Downloaded on January 22,2023 at 10:47:14 UTC from IEEE Xplore. Restrictions apply.

which corresponds to condition 5). Furthermore, $f'(\cdot)$ also satisfies condition 6) since one can always find an upper bound of $f'(\cdot)$ on bounded intervals. The proof then follows by the result of Theorem 3. □

To help establish the relationship between the auxiliary system and our A-SSR system, we first prove the boundedness of the soma current $\boldsymbol{\mu}(t)$ and its average value $\mathbf{u}(t)$.

Lemma 3: There exist an upper bound B_+ and a lower bound B_- such that $\mu_i(t), u_i(t) \in [B_-, B_+], \forall i, t \geq 0$.

Proof: Based on the assumption that all dictionary vectors are normalized to unit Euclidean norm, we define the following auxiliary parameters:

$$\Omega_{\max} \overset{\text{def}}{=} \max_{j \neq i} \left| \boldsymbol{\phi}_i^T \boldsymbol{\phi}_j \right|, \quad b_{\max} \overset{\text{def}}{=} \max_i |b_i| .$$

With the refractory period in spiking neurons, the duration between spikes cannot be arbitrarily small whenever these two spike times exist, *i.e.*, $t_{i,k+1} - t_{i,k} \geq T^{ref}, \forall k \geq 0$. Therefore, the inequality

$$\beta \overset{\text{def}}{=} \frac{1}{\tau} \sum_{c=0}^{\infty} e^{-(cT^{ref}/\tau)} \geq (\alpha * \sigma)(t),$$

holds and the following relationship can be obtained from (15a):

$$\mu_i(t) \leq b_{\max} + (N-1)\, \Omega_{\max} \beta,$$

where N denotes the total number of neurons in A-SSR. Similarly, the relationship $\mu_i(t) \geq -b_{\max} - (N-1)\Omega_{\max}\beta$ holds. Since u_i is the average of μ_i, there exist bounds

$$B_+ = b_{\max} + (N-1)\, \Omega_{\max} \beta,$$
$$B_- = -b_{\max} - (N-1)\Omega_{\max}\beta$$

such that $\mu_i(t), u_i(t) \in [B_-, B_+], \forall i, t \geq 0$. □

Recall that $\nu^- = 0$ is the lower bound of potential $\nu_i(t)$, which means that

$$\int_0^t (\mu_i(s) - \Lambda_i(t))ds - \int_0^t \sigma_i(s)ds \geq 0 \tag{20}$$

holds. By applying operator t^{-1} to (20), we derive the inequality

$$u_i(t) \geq \Lambda_i(t) + a_i(t).$$

Thus $u_i(t) \geq 0$ holds based on $a_i(t) \geq 0$ and $\Lambda_i(t) \geq 0$ from Rule-2 in Section III-B. This implies that $u_i(t)$ can be further bounded by zero in practice, *i.e.*, $u_i(t) \in [0, B_+]$. With the boundedness of $\mu_i(t)$ and $u_i(t)$, we can prove the global asymptotic convergence of our A-SSR system.

Lemma 4: As $t \to \infty$, the firing rate $\mathbf{a}(t)$ of A-SSR globally asymptotically converges to a critical point of the non-convex regularized optimization (13).

Proof: With the derivative derived from (16):

$$\dot{u}_i(t) = \frac{\mu_i(t) - u_i(t)}{t}, \tag{21}$$

the following equation is obtained by applying $t^{-1} \int_0^t ds$ to (11):

$$\tau \dot{u}_i(t) = b_i - u_i(t) - \sum_{j \neq i} \Omega_{ij} a_j(t) - \tau \frac{u_i(t) - \mu_i(0)}{t}. \tag{22}$$

On the other hand, one can derive the following relationship based on (15b) and (15e):

$$a_i(t) = \frac{1}{t} \int_0^t (\mu_i(s) - \Lambda_i(t))ds - \frac{\nu_i(t)}{t}$$
$$= u_i(t) - \Lambda_i(t) - \frac{\nu_i(t)}{t}. \tag{23}$$

Under $a_i(t) \geq 0$, (23) can be further transformed to

$$a_i(t) = \max\left(u_i(t) - \Lambda_i(t) - \frac{\nu_i(t)}{t}, 0 \right). \tag{24}$$

With $\nu_i(t) \in [\nu^-, \nu^s]$ and the result of Lemma 3, it is obvious that (22) and (24) converge to (17) as $t \to \infty$, *i.e.*, A-SSR converges to the auxiliary system. Based on Lemma 2, the $\mathbf{u}(t)$ and $\mathbf{a}(t)$ of A-SSR are also globally asymptotically convergent when time goes to infinity.

Assume that $\mathbf{u}^*$ is the stable point of A-SSR and $\mathbf{a}^*$ is the corresponding output. With (21) and the result of Lemma 3, we have

$$\dot{\mathbf{u}}^* = \lim_{t \to \infty} \dot{\mathbf{u}}(t) = \lim_{t \to \infty} \frac{\boldsymbol{\mu}(t) - \mathbf{u}(t)}{t} = 0. \tag{25}$$

Since A-SSR converges to the auxiliary system, our A-SSR system satisfies $\tau \dot{\mathbf{u}}(t) \in -\partial E(\mathbf{a}(t))$ as $t \to \infty$. According to (25), it is clear that

$$0 \in \partial E(\mathbf{a}^*)$$

holds, meaning that as time goes to infinity, the firing rate $\mathbf{a}(t)$ converges to $\mathbf{a}^*$, a critical point of (13). □

With the above results, we complete the proof of Theorem 2.

ACKNOWLEDGMENT

The authors would like to thank the support of Intel neuromorphic research community project (project ID: RV2.123.Yu and RV3.124.Yu) with the Loihi chip used for this research.

REFERENCES

[1] Y. C. Eldar and G. Kutyniok, *Compressed Sensing : Theory and Applications*. Cambridge, U.K.: Cambridge Univ. Press, 2012.
[2] I. Daubechies, M. Defrise, and C. De Mol, "An iterative thresholding algorithm for linear inverse problems with a sparsity constraint," *Commun. Pure Appl. Math.*, vol. 57, no. 11, pp. 1413–1457, Nov. 2004.
[3] S. S. Chen, D. L. Donoho, and M. A. Saunders, "Atomic decomposition by basis pursuit," *SIAM Rev.*, vol. 43, no. 1, pp. 129–159, 2001.
[4] R. Tibshirani, "Regression shrinkage and selection via the lasso," *J. Roy. Stat. Soc.: Ser. B. (Methodological)*, vol. 58, no. 1, pp. 267–288, Jan. 1996.
[5] S. Ghosh-Dastidar and H. Adeli, "Spiking neural networks," *Int. J. Neural Syst.*, vol. 19, no. 4, pp. 295–308, 2009.
[6] W. Maass, "Lower bounds for the computational power of networks of spiking neurons," *Neural Comput.*, vol. 8, no. 1, pp. 1–40, 1996.
[7] W. Maass and C. Bishop, *Pulsed Neural Networks*. Cambridge, MA, USA: MIT Press, Jan. 1998.
[8] T.-H. Lin and P. T. P. Tang, "Sparse dictionary learning by dynamical neural networks," in *Proc. Int. Conf. Learn. Representations*, 2018, pp. 1–12.
[9] S. Shapero, C. Rozell, and P. Hasler, "Configurable hardware integrate and fire neurons for sparse approximation," *Neural Netw.*, vol. 45, pp. 134–143, Sep. 2013.
[10] S. Shapero, M. Zhu, J. Hasler, and C. Rozell, "Optimal sparse approximation with integrate and fire neurons," *Int. J. Neural Syst.*, vol. 24, no. 05, Aug. 2014, Art. no. 1440001.
[11] P. T. P. Tang, T.-H. Lin, and M. Davies, "Sparse coding by spiking neural networks: Convergence theory and computational results," 2017, *arXiv:1705.05475*.

Authorized licensed use limited to: Weizmann Institute of Science. Downloaded on January 22,2023 at 10:47:14 UTC from IEEE Xplore. Restrictions apply.

[12] J. Zylberberg, J. T. Murphy, and M. R. DeWeese, "A sparse coding model with synaptically local plasticity and spiking neurons can account for the diverse shapes of V1 simple cell receptive fields," *PLoS Comput. Biol.*, vol. 7, no. 10, Oct. 2011, Art. no. e1002250.
[13] T. Hu, A. Genkin, and D. B. Chklovskii, "A network of spiking neurons for computing sparse representations in an energy-efficient way," *Neural Comput.*, vol. 24, no. 11, pp. 2852–2872, 2012.
[14] E. J. Candès, M. B. Wakin, and S. P. Boyd, "Enhancing sparsity by reweighted ℓ_1 minimization," *J. Fourier Anal. Appl.*, vol. 14, no. 5-6, pp. 877–905, Dec. 2008.
[15] G. Gasso, A. Rakotomamonjy, and S. Canu, "Recovering sparse signals with a certain family of nonconvex penalties and DC programming," *IEEE Trans. Signal Process.*, vol. 57, no. 12, pp. 4686–4698, Dec. 2009.
[16] I. Daubechies, R. DeVore, M. Fornasier, and C. S. Güntürk, "Iteratively reweighted least squares minimization for sparse recovery," *Commun. Pure Appl. Math.*, vol. 63, no. 1, pp. 1–38, Jan. 2010.
[17] Z. Lu, "Iterative reweighted minimization methods for l_p regularized unconstrained nonlinear programming," *Math. Program.*, vol. 147, no. 1, pp. 277–307, 2014.
[18] W. Cao, J. Sun, and Z. Xu, "Fast image deconvolution using closed-form thresholding formulas of regularization," *J. Vis. Commun. Image Representation*, vol. 24, no. 1, pp. 31–41, Jan. 2013.
[19] Y. Qian, S. Jia, J. Zhou, and A. Robles-Kelly, "Hyperspectral unmixing via $L_{1/2}$ sparsity-constrained nonnegative matrix factorization," *IEEE Trans. Geosci. Remote Sens.*, vol. 49, no. 11, pp. 4282–4297, Nov. 2011.
[20] C. J. Rozell, D. H. Johnson, R. G. Baraniuk, and B. A. Olshausen, "Sparse coding via thresholding and local competition in neural circuits," *Neural Comput.*, vol. 20, no. 10, pp. 2526–2563, Oct. 2008.
[21] A. S. Charles, P. Garrigues, and C. J. Rozell, "A common network architecture efficiently implements a variety of sparsity-based inference problems," *Neural Comput.*, vol. 24, no. 12, pp. 3317–3339, Dec. 2012.
[22] M. Davies et al., "Loihi: A neuromorphic manycore processor with on-chip learning," *IEEE Micro*, vol. 38, no. 1, pp. 82–99, Jan./Feb. 2018.
[23] A. Beck and M. Teboulle, "A fast iterative shrinkage-thresholding algorithm for linear inverse problems," *SIAM J. Imag. Sci.*, vol. 2, no. 1, pp. 183–202, Jan. 2009.
[24] K. Roy, A. Jaiswal, and P. Panda, "Towards spike-based machine intelligence with neuromorphic computing," *Nature*, vol. 575, no. 7784, pp. 607–617, 2019.
[25] K. L. Fair, D. R. Mendat, A. G. Andreou, C. J. Rozell, J. Romberg, and D. V. Anderson, "Sparse coding using the locally competitive algorithm on the TrueNorth neurosynaptic system," *Front. Neurosci.*, vol. 13, pp. 1–15, Jul. 2019.
[26] E. M. Izhikevich, "Which model to use for cortical spiking neurons?," *IEEE Trans. Neural Netw.*, vol. 15, no. 5, pp. 1063–1070, Sep. 2004.
[27] A. L. Hodgkin and A. F. Huxley, "A quantitative description of membrane current and its application to conduction and excitation in nerve," *J. Physiol.*, vol. 117, no. 4, pp. 500–544, 1952.
[28] W. Gerstner, W. M. Kistler, R. Naud, and L. Paninski, *Neuronal Dynamics: From Single Neurons to Networks and Models of Cognition*. Cambridge, U.K.: Cambridge Univ. Press, 2014.
[29] R. Vaquer-Sunyer et al., "A million spiking-neuron integrated circuit with a scalable communication network and interface," *Science*, vol. 345, no. 6197, pp. 668–673, 2014.
[30] S. B. Furber, F. Galluppi, S. Temple, and L. A. Plana, "The SpiNNaker project," *Proc. IEEE*, vol. 102, no. 5, pp. 652–665, May 2014.
[31] S. Moradi, N. Qiao, F. Stefanini, and G. Indiveri, "A scalable multicore architecture with heterogeneous memory structures for dynamic neuromorphic asynchronous processors (DYNAPs)," *IEEE Trans. Biomed. Circuits Syst.*, vol. 12, no. 1, pp. 106–122, Feb. 2018.
[32] M. Davies et al., "Advancing neuromorphic computing with loihi: A survey of results and outlook," *Proc. IEEE*, vol. 109, no. 5, pp. 911–934, May 2021.
[33] J. Pei et al., "Towards artificial general intelligence with hybrid tianjic chip architecture," *Nature*, vol. 572, no. 7767, pp. 106–111, Aug. 2019.
[34] B. K. Natarajan, "Sparse approximate solutions to linear systems," *SIAM J. Comput.*, vol. 24, no. 2, pp. 227–234, Apr. 1995.
[35] J. J. Hopfield, "Neurons with graded response have collective computational properties like those of two-state neurons," *Proc. Nat. Acad. Sci.*, vol. 81, no. 10, pp. 3088–3092, 1984.
[36] A. Balavoine, C. J. Rozell, and J. Romberg, "Convergence of a neural network for sparse approximation using the nonsmooth Łojasiewicz inequality," in *Proc. IEEE Int. Joint Conf. Neural Netw.*, 2013, pp. 1–8.
[37] S. Shapero, A. S. Charles, C. J. Rozell, and P. Hasler, "Low power sparse approximation on reconfigurable analog hardware," *IEEE J. Emerg. Sel. Topics Circuits Syst.*, vol. 2, no. 3, pp. 530–541, Sep. 2012.
[38] G. W. Roberts and V. W. Leung, *Design and Analysis of Integrator-Based Log-Domain Filter Circuits*, vol. 534. Berlin, Germany: Springer Science & Business Media, 1999.
[39] L. Chen and Y. Gu, "The convergence guarantees of a non-convex approach for sparse recovery," *IEEE Trans. Signal Process.*, vol. 62, no. 15, pp. 3754–3767, Aug. 2014.
[40] T. Bekolay et al., "Nengo: A python tool for building large-scale functional brain models," *Front. Neuroinform.*, vol. 7, pp. 48–48, 2014.
[41] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proc. IEEE*, vol. 86, no. 11, pp. 2278–2324, Nov. 1998.
[42] J. Mairal, "Sparse modeling for image and vision processing," *Found. Trends Comput. Graph. Vis.*, vol. 8, no. 2-3, pp. 85–283, 2014.
[43] Z. Wang, E. P. Simoncelli, and A. C. Bovik, "Multiscale structural similarity for image quality assessment," in *Proc. IEEE 37th Asilomar Conf. Sign. Syst. Comput.*, 2003, vol. 2, pp. 1398–1402.
[44] J. Wright, A. Y. Yang, A. Ganesh, S. S. Sastry, and Y. Ma, "Robust face recognition via sparse representation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 2, pp. 210–227, Feb. 2009.
[45] A. S. Georghiades, P. N. Belhumeur, and D. J. Kriegman, "From few to many: Illumination cone models for face recognition under variable lighting and pose," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 23, no. 6, pp. 643–660, Jun. 2001.
[46] L. Yu, G. Zheng, and J. P. Barbot, "Dynamical sparse recovery with finite-time convergence," *IEEE Trans. Signal Process.*, vol. 65, no. 23, pp. 6146–6157, Dec. 2017.

Xiang Zhang received the B.E. degree in communication engineering in 2020 from Wuhan University, Wuhan, China, where he is currently working toward the M.S. degree in information and communication engineering with the electronic information school. His research interests include computer vision and neuromorphic computation.

Lei Yu (Member, IEEE) received the B.S. and Ph.D. degrees in signal processing from Wuhan University, Wuhan, China, in 2006 and 2012, respectively. From 2013 to 2014, he was a Postdoc Researcher with the VisAGeS Group, Institut National de Recherche en Informatique et en Automatique (INRIA) for one and a half years. He is currently an Associate Professor with the School of Electronics and Information, Wuhan University. From 2016 to 2017, he was also a Visiting Professor with Duke University, Durham, NC, USA. He was a Guest Professor with the École Nationale Supérieure de l'Électronique et de ses Applications (ENSEA), Cergy, France, for one month in 2018. His research interests include event-based vision, neuromorphic computation, and signal processing.

Authorized licensed use limited to: Weizmann Institute of Science. Downloaded on January 22,2023 at 10:47:14 UTC from IEEE Xplore. Restrictions apply.

Gang Zheng (Senior Member, IEEE) received the B.E. and M.E. degrees in communication and systems from Wuhan University, Wuhan, China, in 2001 and 2004, respectively, and the Ph.D. degree in automatic control from École Nationale Supérieure de l'Électronique et de ses Applications (ENSEA), Cergy-Pontoise, France, in 2006. Since 2007, he has been a Postdoctor with INRIA Grenoble, Laboratoire Jean Kuntzmann, ENSEA. In September 2009, he joined INRIA Lille, Lille, France, as a Permanent Researcher. His research interests include control and observation/estimation of nonlinear systems and their applications to robotics.

Yonina C. Eldar (Fellow, IEEE) received the B.Sc. degree in physics and the second B.Sc. degree in electrical engineering from Tel-Aviv University (TAU), Tel-Aviv, Israel, in 1995 and 1996, respectively, and the Ph.D. degree in electrical engineering and computer science from the Massachusetts Institute of Technology, Cambridge, MA, USA, in 2002. She is currently a Professor with the Department of Mathematics and Computer Science, Weizmann Institute of Science, Rehovot, Israel. She was previously a Professor with the Department of Electrical Engineering, Technion. She is also a Visiting Professor with MIT, a Visiting Scientist with Broad Institute, and an Adjunct Professor with Duke University, Durham, NC, USA, and was a Visiting Professor with Stanford. She is the author of the book *Sampling Theory: Beyond Bandlimited Systems* and co-author of five other books published by Cambridge University Press. Her research interests include statistical signal processing, sampling theory and compressed sensing, learning and optimization methods, and their applications to biology, and medical imaging and optics. She is a Member of the Israel Academy of Sciences and Humanities (elected 2017), and a EURASIP Fellow.

Dr. Eldar was the recipient of many awards for excellence in research and teaching, including the IEEE Signal Processing Society Technical Achievement Award in 2013, IEEE/AESS Fred Nathanson Memorial Radar Award in 2014, IEEE Kiyo Tomiyasu Award in 2016, Michael Bruno Memorial Award from the Rothschild Foundation, Weizmann Prize for Exact Sciences, Wolf Foundation Krill Prize for Excellence in Scientific Research, Henry Taub Prize for Excellence in Research (twice), Hershel Rich Innovation Award (three times), Award for Women with Distinguished Contributions, Andre and Bella Meyer Lectureship, Career Development Chair at the Technion, Muriel & David Jacknow Award for Excellence in Teaching, and Technion's Award for Excellence in Teaching (two times). She was a Horev Fellow of the Leaders in Science and Technology Program at the Technion and an Alon Fellow. She was also the recipient of several best paper awards and best demo awards together with her research students and colleagues, including the SIAM outstanding Paper Prize, UFFC Outstanding Paper Award, Signal Processing Society Best Paper Award, and IET Circuits, Devices and Systems Premium Award. She was selected as one of the 50 most influential women in Israel and in Asia, and is a highly cited researcher.

She was a Member of the Young Israel Academy of Science and Humanities and Israel Committee for Higher Education. She is the Editor in Chief of Foundations and Trends in Signal Processing, a Member of the IEEE Sensor Array and Multichannel Technical Committee and serves on several other IEEE committees. In the past, she was a Signal Processing Society Distinguished Lecturer, Member of the IEEE Signal Processing Theory and Methods and Bio Imaging Signal Processing technical committees, and was an Associate Editor for the IEEE Transactions On Signal Processing, *EURASIP Journal of Signal Processing, SIAM Journal on Matrix Analysis and Applications*, and *SIAM Journal on Imaging Sciences*. She was the Co-Chair and Technical Co-Chair of several international conferences and workshops.

Authorized licensed use limited to: Weizmann Institute of Science. Downloaded on January 22,2023 at 10:47:14 UTC from IEEE Xplore. Restrictions apply.