Versatile emulation of spiking neural networks on an accelerated neuromorphic substrate

S. Billaudelle*[1], Y. Stradmann*[1], K. Schreiber*[1], B. Cramer*[1], A. Baumbach*[1], D. Dold*[1], J. Göltz*[1], A. F. Kungl*[1], T. C. Wunderlich*[3 1], A. Hartel[1], E. Müller[1], O. Breitwieser[1], C. Mauch[1], M. Kleider[1], A. Grübl[1], D. Stöckel[1], C. Pehle[1], A. Heimbrecht[1], P. Spilger[1], G. Kiene[1], V. Karasenko[1], W. Senn[2], M. A. Petrovici*[§ 2 1], J. Schemmel[§ 1] and K. Meier[§ 1 †]

*Authors with equal contribution §Shared senior authorship † Deceased
[1]Kirchhoff-Institute for Physics, Heidelberg University [2]Department of Physiology, University of Bern
[3]Berlin Institute of Health, Berlin / Charité–Universitätsmedizin, Berlin

***Abstract*—We present first experimental results on the novel BrainScaleS-2 neuromorphic architecture based on an analog neuro-synaptic core and augmented by embedded microprocessors for complex plasticity and experiment control. The high acceleration factor of 1000 compared to biological dynamics enables the execution of computationally expensive tasks, by allowing the fast emulation of long-duration experiments or rapid iteration over many consecutive trials. The flexibility of our architecture is demonstrated in a suite of five distinct experiments, which emphasize different aspects of the BrainScaleS-2 system.**

I. Introduction

The unifying principle behind all neuromorphic architectures lies in their attempt to emulate certain structural and dynamical aspects of biological nervous systems in order to inherit some of their well-known functional and metabolic advantages over conventional silicon substrates. However, precisely what these aspects are remains a holy grail of computational neuroscience, and our best attempts at answering this question are still rather conjectural. This state of active exploration is reflected by the broad diversity of the current neuromorphic landscape [1].

Consequently, our approach to neuromorphic engineering is explicitly geared towards building systems that can serve as scientific tools for studying this question. By adhering to a restricted set of biologically inspired principles, we enable an efficient implementation in silico with respect to both emulation speed and power consumption. Additionally, our proposed architecture emphasizes precision, scalability and, in particular, a substantial degree of flexibility. This relates not only to the wide-ranged configurability of neuro-synaptic parameters and connectivity, but most importantly to the ability of influencing our circuits during emulation, which goes significantly beyond synaptic plasticity, as outlined below.

In this manuscript, we describe the core principles underlying the BrainScaleS-2 architecture, followed by the emulation of a diverse set of spiking neural networks, which we have chosen to emphasize different aspects of the chip's operation and capabilities, as well as different computational principles that we believe are relevant for biological information processing.

II. Architecture

BrainScaleS-2 is a family of mixed-signal neuromorphic systems implemented in a 65 nm node. It is centered around an analog neural network core implementing biologically inspired neuron and synapse circuits (Fig. 1).

The neurons [2] feature leaky integrate-and-fire (LIF) dynamics with synaptic currents modeled as superpositions of spike-triggered exponential kernels. The membrane is connected by a programmable conductance to a reset potential for a finite refractory period as soon as it crosses a certain threshold. Additional mechanisms such as neuronal adaptation and exponential near-threshold dynamics [3] or dendritic interaction [4] enable the emulation of more complex structure and dynamics. All neurons are individually configurable via an on-chip analog parameter memory [5] and a set of digital control values.

Voltages and currents are – scaled to utilize the available dynamic range – directly represented in the respective circuits and evolve in continuous time. Leveraging the intrinsic capacitances and conductances of the technology, time constants of neuron and synapse dynamics are rendered 1000 times smaller compared to typical values found in biology. This thousand-fold acceleration facilitates the execution of time-consuming tasks, such as performing high-dimensional parameter sweeps, the investigation of learning and metalearning, or statistical computations requiring large volumes of data [6], [7].

Each neuron is associated with a column of synapse circuits [8], which receive their inputs from the chip's digital backend. The 6 bit synaptic weight is stored in local static random-access memory (SRAM). It further holds a 6 bit label, enabling synapses to filter afferent events tagged with their respective source address. Each synapse also implements an analog circuit for measuring pairwise correlations between pre- and post-synaptic spike events [8], enabling access to various forms of spike-timing-dependent plasticity (STDP). The analog correlation traces are accessible via column-parallel analog-to-digital converters (CADCs), which also allow the digitization of neuronal observables such as the membrane potential.

The versatility of the BrainScaleS-2 architecture is substantially augmented by the incorporation of freely programmable

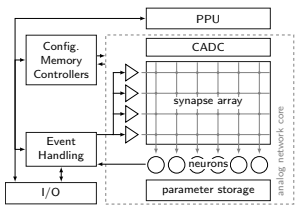

Fig. 1. **Simplified block-level schematic of a BrainScaleS-2 ASIC.** The analog neuromorphic core is surrounded by event transport logic and control logic, including controllers for full-custom configuration SRAM. Details and components that lie beyond the scope of this paper were omitted.

978-1-7281-3320-1/20/$31.00 ©2020 IEEE

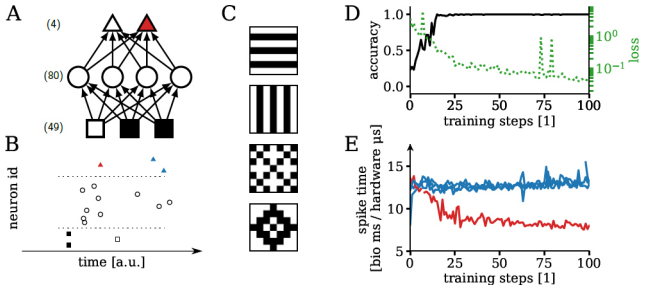

Fig. 2. **Pattern recognition with time-to-first-spike coding. A)** Hierarchical network structure and neuron numbers per layer. **B)** Input (□) spike times encode input pixel brightness. Activity propagates through the hidden layer (○) to the label layer (△). There, classification is determined by the identity of the first neuron to spike (red). **C)** Training/test set consisting of four patterns. **D)** Accuracy increase and corresponding decrease of loss during learning. **E)** Evolution of label neuron spike times during training for one example image.

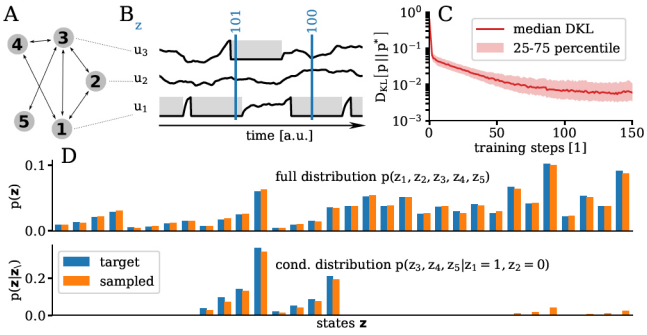

Fig. 3. **Spike-based Bayesian inference. A)** Schematic of a random spiking sampling network. **B)** Membrane voltages of three selected neurons and visualisation of the spike-based representation of binary random variables. **C)** Sampling performance after training for 500 randomly generated target distributions. **D)** Sampling from the learned (top) and an associated conditional distribution (bottom). Orange: sampled distribution. Blue: analytically calculated target distribution. Remaining error bars are too small to visualize.

embedded microprocessors [8]. Together with their single instruction, multiple data (SIMD) vector units, which are tightly coupled to the synapse arrays' SRAM controllers and the CADCs, they form plasticity processing units (PPUs) for efficient control of synaptic plasticity. Access to the on-chip configuration bus further allows the processor to also reconfigure all other components of the neuromorphic system during experiment execution. The PPU can thus be used for a vast array of applications such as near-arbitrary learning rules, on-line circuit calibration, structural network reconfiguration, or the co-simulation of an environment capable of continuous interaction with the network running on the neuromorphic core.

The experiments below were implemented on two revisions of the BrainScaleS-2 architecture. Sections III-B through III-E cover experiments conducted on a prototype featuring 32 neurons and 32×32 synapses. A full-size system with 512 neurons and 512×256 synapses was used for III-A.

III. Experiments

A. Deep learning using precise spike timing

In many applications, the time and energy to solution represent essential commodities. For spiking networks, optimal use of these resources often imposes to have as few and as early spikes as possible. However, the discrete nature of spikes makes it difficult to apply conventional machine learning algorithms based on differentiable loss functions.

In the time-to-first-spike coding scheme, a neuron encodes a continuous variable as the time elapsed before its first spike. The decision of a network performing a classification task is given by the first neuron to spike in the label layer (Fig. 2A,B). For such networks, an efficient gradient-descent-based learning scheme was first proposed in [9], using error backpropagation on a continuous function of output spike times.

We have generalized this method to include an exact, closed-form expression for finite membrane time constants [10], [11] and applied it to a 3-layer network emulated on BrainScaleS-2 (Fig. 2). The loss was calculated as the cross-entropy of a softmax function on negative spike times, in order to maximize the distance between correct and incorrect label layer spikes. Fig. 2D shows its evolution during the host-based training and the associated classification accuracy for a simple 4-class learning task (Fig. 2C). The evolution of the label neuron spike times for one example class is shown in Fig. 2E. The robustness of the applied learning rule and the emulated network dynamics is evidenced by the clear separation of first-spike times.

A particularly appealing feature of this implementation is its extreme communication sparsity, with only one input spike per input variable and at most one spike per emulated neuron before classification. After learning, the emulated network needed less than 10 µs to classify an image. This duration scales proportionally to the chosen synaptic and membrane time constants, which in our case were set to 5 µs. Taking into consideration relaxation times between patterns, our setup is able to handle a pattern throughput of at least 20 kHz, independently of emulated network size [11].

B. Sampling-based Bayesian computation

The Bayesian brain hypothesis [12] aspires to explain how the mammalian brain can operate in a probabilistic sea of sensory data. In [13], it was shown how networks of LIF neurons can learn to perform Bayesian inference through sampling on high-dimensional data distributions [14], [15].

In this quintessentially spike-based framework, neurons become stochastic due to background spiking input, thereby lending themselves to the representation of binary random variables: during post-spike refractoriness, a neuron is considered to be in the state $z = 1$, and $z = 0$ otherwise (Fig. 3A,B). With appropriate synaptic connections, the resulting network dynamics inherently generate a sequence of samples from the learned distribution. This enables the training of spiking networks to perform sampling-based Bayesian inference in arbitrary binary probability spaces, with applications to generative as well as discriminative problems [16], [17].

Contrastive Hebbian learning [18] was performed with the hardware in the loop, i.e., with updates calculated on a host PC [17], [19]. Each training step was run for 100 ms, corresponding to 100 s bio time and approximately 5000 samples.

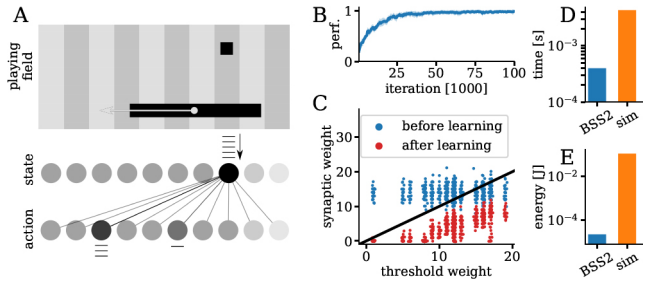

Fig. 4. **Reinforcement learning with reward-modulated STDP. A)** The PPU simulates a simplified version of Pong. The horizontal position of the ball serves as input for a 2-layer neural network, with the resulting output dictating the target paddle position. The network receives reward based on its aiming accuracy. **B)** Playing performance during learning. **C)** Synaptic depression automatically adapts to the excitability of neurons. **D, E)** Wall-clock duration and power consumption of a single iteration on BrainScaleS-2 (blue) and an equivalent software simulation using NEST (orange).

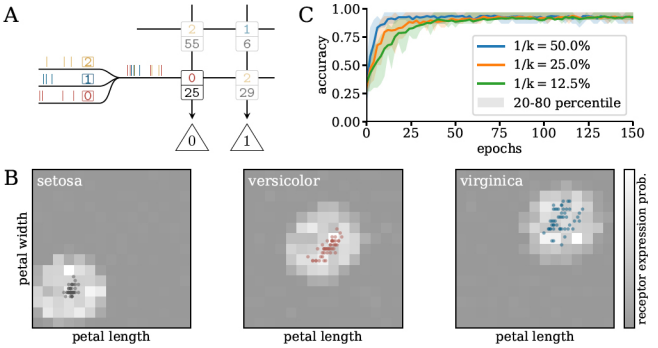

Fig. 5. **Self-organizing receptive fields through structural plasticity. A)** Spike trains from different sources can be injected into a single synaptic row. Each synapse filters afferent spikes according to a locally stored label. **B)** A network endowed with structural plasticity learns to discriminate between types of Iris flowers (dataset represented by colored dots). The receptor distribution after training is adapted to the input data distribution. **C)** Feature selection through structural plasticity allows the conservation of classification performance even for strongly enforced sparsity $1 - 1/k$.

Training was monitored using the Kullback-Leibler divergence between sampled and target distribution (Fig. 3C). After training, the network reliably sampled from its target distribution and from associated conditional distributions (Bayesian inference, Fig. 3D). Compared to previous neuromorphic realizations of neural sampling with analog neurons [17], [20], the BrainScaleS-2 system allows unprecedented precision, while still enabling fast inference due to its thousandfold acceleration.

C. Reinforcement learning

Recent advances in reinforcement learning have enabled artificial systems to achieve unprecedented performance in board and computer games [21]. Having clear roots in neurobiology, it is also an interesting framework for neuromorphic agents that learn through repeated interaction with an environment.

Three-factor learning rules [22] can implement reinforcement learning in spiking neural networks using a global neuromodulator and local observables such as spike rates. As already shown in [23], the BrainScaleS-2 architecture supports the implementation of an R-STDP learning rule [22], [24] in a closed-loop setup contained fully on chip. Its application to a simplified version of the Pong video game is shown in Fig. 4A. The network dynamics were emulated by the neuromorphic substrate, while the embedded plasticity processor took on a dual role. First, it simulated the game dynamics, creating a host-independent setup. Second, it calculated the plasticity updates using the synaptically stored correlation traces according to $\Delta w_{ij} \propto \left(R - \bar{R}\right) e_{ij}$, where R is the reward, $\bar{R}$ its moving average and e_{ij} an STDP-like eligibility trace.

During training, the network learned to keep the ball close to the middle of the paddle (Fig. 4B). Implicitly, the experiment also demonstrates how learning can compensate analog fixed-pattern noise (Fig. 4C): while the excitability of uncalibrated neurons varied significantly due to mismatch effects, synapses that would negatively impact correct tracking of the ball were systematically depressed to a subthreshold strength with respect to their postsynaptic neuron. Furthermore, this setup demonstrates the speed and power advantages of the BrainScaleS-2 architecture compared to software simulations (Fig. 4D/E).

D. Structural plasticity

Synaptic plasticity is well known to not only be limited to adjusting the strength of synapses; the connectome itself undergoes continuous change during the lifetime of an individual [25]–[27]. By constraining the number of expressed synapses to enforce a certain level of sparsity, the nervous system appears to manage its spatial and energetic budget [28]. Similar constraints apply to all physical information-processing systems, with neuromorphic ones being no exception. In particular, the synaptic fan-in of silicon neurons is often limited.

We implemented a synaptic update policy that incorporates structural plasticity, enabling neurons to dynamically select a set of suitable synapses out of a pool of potential connections, that optimizes performance for a chosen task while maintaining a sparse connectome [29]. The learning rule is composed of three parts: an STDP term that potentiates correlated connections, a homeostatic regularizer that limits post-synaptic firing rates and encourages synaptic competition, and a stochastic component that induces exploration. A pruning condition is executed periodically, removing synapses with a weight below a certain threshold and randomly reassigning them.

Structural plasticity is enabled by bundling k presynaptic sources and injecting them into a single synapse row (Fig. 5A): as each synapse can only gate one of these to its home neuron, pruning and reassigning of a synapse is simply implemented by changing its label. The reconfiguration is thereby fully local and, in particular, does not involve time-consuming sorting of routing tables or connectivity lists [30]. If bundles are disjoint, their size k also effectively sets the synaptic sparsity to $1-1/k$.

We applied the above algorithm to a supervised learning task, where the network was trained to classify the Iris data set [31]. We randomly placed 48 receptor neurons on the two-dimensional feature plane spanned by petal width and length. The firing rate of a receptor was set to increase with its proximity to a presented data point. In three separate scenarios, the resulting $n = 48$ input spike trains were injected

into $m = 6$, 12, and 24 synaptic rows, leading to three different levels of sparsity: each label neuron could only see $1/k = m/n = 12.5\,\%$, $25.0\,\%$, and $50.0\,\%$ of the receptors at each point in time, respectively. During training, teacher stimuli ensured that the correct label neurons were excited when an input belonging to their respective class was presented.

The emulated plasticity rule led to self-organized reconfiguration of their receptive fields (Fig. 5B), as the correlation between teacher signal and receptor proximity to the presented data drove the potentiation of associated synaptic weights. For higher degrees of enforced sparsity, convergence times were longer, as the search for relevant inputs in the feature space became statistically more challenging. Ultimately however, the learning rule enabled the network to achieve near-perfect classification in all three scenarios (Fig. 5C), demonstrating its ability to ensure a better utilization of synaptic resources without prior knowledge of the input data.

E. Insect navigation

Recent developments in biological imaging and data processing have facilitated unprecedented insight into numerous functional aspects of insect brains [32]–[34]. For example, it has been shown that a structure known as the central complex is involved in navigational behavior [35]. Based on physiological data from the bee's central complex and following [36], we emulated a network for path integration (Fig. 6A) that reproduces bees' ability to return to their nest's location after exploring the environment for sources of food.

Each experiment started with a spread-out phase, in which a virtual insect performed a random walk starting from a certain origin. During this phase, the modeled network received the sensory data of the absolute head orientation and the optical flow field of a left and right eye, but had no effect on the insect motion. In the second part of the experiment, the return phase, the insect's motion was determined by two motor neurons within the network. The insect's head orientation was encoded by four spike sources, each representing a cardinal direction similar to a compass. The optical flow field was similarly represented by two spike generators that fired with a rate proportional to the optical flow as derived from the left and right eye, respectively (FL, FR). Moreover, the two motor neurons (ML, MR) steered the insect by providing propulsion on the left or right hand side, similar to a tank drive.

While the model in [36] comprises 90 fire-rate-neurons with floating-point precision, the network on BrainScaleS-2 achieved about the same functional performance with only 18 neurons. Additionally, we implemented the short-term memory mechanism employed by the integrator neurons to store directional distance as a synaptic mechanism.

The total flight duration was set to 200 ms on the hardware, which corresponds to 200 s in biology. In that time, sensory information and steering signals were exchanged between body and brain every 100 µs. During the first 50 ms the insect performed a random outbound journey, after which it returned to the nest. Sample trajectories can be seen in Fig. 6B,C. The average spike rate of all neurons and spike generators

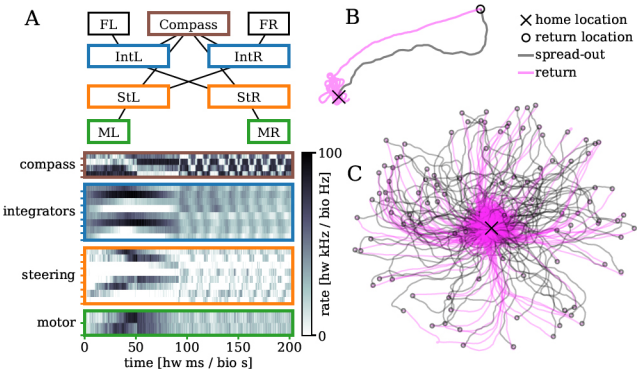

Fig. 6. **Virtual insectoid agent performing path integration on BrainScaleS-2. A)** Network schematic and activity histogram. The information flows from the sensory layer at the top through an integration and a steering layer to the motor neurons at the bottom. R and L indicate the right and left side, respectively. **B)** A typical trajectory of the virtual insect which turns to random looping around the home position upon reaching it. **C)** Overlay of 100 trajectories like in B), each with a different random outbound journey.

was 30 kHz (30 Hz bio), which is in good agreement with experimental data from drosophila [37] or locusts [38].

In this experiment, the PPU handled multiple tasks: the processing of synaptic modulations for the integrator neurons, the simulation of the environment, an emulation of all sensors including the corresponding spike stimuli, the translation of neuronal data into actions of motion, and the entire experiment control. Apart from the setup and readout phase, the experiment ran entirely self-contained on the BrainScaleS-2 system.

IV. Discussion and Outlook

In a post-Moore era, neuromorphic circuits represent a promising venue for advancing the computational capabilities of silicon. This manuscript motivates how, by coupling the advantages of analog circuits with the flexibility of general-purpose digital computation and control, our BrainScaleS-2 architecture contributes to the research-oriented territory of the neuromorphic landscape. In our endeavor, we share a common goal with other promising architectures such as [39] and [40], which follow radically different design paradigms with advantages and drawbacks of their own.

A key aspect that we do not address above, but ultimately decides the value of such systems for computational neuroscience research, is their scalability. Following integration concepts first proposed in [41] and studied in, e.g., [19], [42], the BrainScaleS-2 architecture is explicitly designed to scale up to large, multi-chip systems. These will conserve the network-size-independent speedup and energy efficiency that we have addressed in our above experiments, thus providing access to spiking network studies that are otherwise prohibitive for simulation software running on conventional substrates.

Acknowledgements

We gratefully acknowledge funding from the European Union under grant agreements 604102, 720270, 785907 (HBP) and the Manfred Stärk Foundation.

REFERENCES

[1] C. S. T. Thakur, J. Molin, G. Cauwenberghs, G. Indiveri, K. Kumar, N. Qiao, J. Schemmel, R. M. Wang *et al.*, "Large-scale neuromorphic spiking array processors: A quest to mimic the brain," *Frontiers in neuroscience*, vol. 12, p. 891, 2018.

[2] S. A. Aamir, Y. Stradmann, P. Müller, C. Pehle, A. Hartel, A. Grübl, J. Schemmel, and K. Meier, "An accelerated lif neuronal network array for a large-scale mixed-signal neuromorphic architecture," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 65, no. 12, pp. 4299–4312, Dec 2018.

[3] S. A. Aamir, P. Müller, G. Kiene, L. Kriener, Y. Stradmann, A. Grübl, J. Schemmel, and K. Meier, "A mixed-signal structured adex neuron for accelerated neuromorphic cores," *IEEE Transactions on Biomedical Circuits and Systems*, vol. 12, no. 5, pp. 1027–1037, Oct 2018.

[4] J. Schemmel, L. Kriener, P. Müller, and K. Meier, "An accelerated analog neuromorphic hardware system emulating nmda-and calcium-based non-linear dendrites," in *2017 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2017, pp. 2217–2226.

[5] M. Hock, A. Hartel, J. Schemmel, and K. Meier, "An analog dynamic memory array for neuromorphic hardware," in *Circuit Theory and Design (ECCTD), 2013 European Conference on*, Sep. 2013, pp. 1–4.

[6] B. Cramer, D. Stöckel, M. Kreft, J. Schemmel, K. Meier, and V. Priesemann, "Control of criticality and computation in spiking neuromorphic networks with plasticity," *arXiv preprint arXiv:1909.08418*, 2019.

[7] T. Bohnstingl, F. Scherr, C. Pehle, K. Meier, and W. Maass, "Neuromorphic hardware learns to learn," *Frontiers in neuroscience*, vol. 13, 2019.

[8] S. Friedmann, J. Schemmel, A. Grübl, A. Hartel, M. Hock, and K. Meier, "Demonstrating hybrid learning in a flexible neuromorphic hardware system," *IEEE Transactions on Biomedical Circuits and Systems*, vol. 11, no. 1, pp. 128–142, 2017.

[9] H. Mostafa, "Supervised learning based on temporal coding in spiking neural networks," *IEEE transactions on neural networks and learning systems*, vol. 29, no. 7, pp. 3227–3235, 2017.

[10] J. Göltz, "Training deep networks with time-to-first-spike coding on the brainscales wafer-scale system," Masterarbeit, Universität Heidelberg, April 2019. [Online]. Available: http://www.kip.uni-heidelberg.de/Veroeffentlichungen/details.php?id=3909

[11] J. Göltz, A. Baumbach, S. Billaudelle, O. Breitwieser, D. Dold, L. Kriener, A. F. Kungl, W. Senn *et al.*, "Fast and deep neuromorphic learning with time-to-first-spike coding," *arXiv preprint arXiv:1912.11443*, 2019.

[12] K. Doya, S. Ishii, A. Pouget, and R. P. Rao, *Bayesian brain: Probabilistic approaches to neural coding*. MIT press, 2007.

[13] M. A. Petrovici, J. Bill, I. Bytschok, J. Schemmel, and K. Meier, "Stochastic inference with spiking neurons in the high-conductance state," *Physical Review E*, vol. 94, no. 4, p. 042312, 2016.

[14] L. Leng, R. Martel, O. Breitwieser, I. Bytschok, W. Senn, J. Schemmel, K. Meier, and M. A. Petrovici, "Spiking neurons with short-term synaptic plasticity form superior generative networks," *Scientific reports*, vol. 8, no. 1, p. 10651, 2018.

[15] D. Dold, I. Bytschok, A. F. Kungl, A. Baumbach, O. Breitwieser, W. Senn, J. Schemmel, K. Meier, and M. A. Petrovici, "Stochasticity from function—why the bayesian brain may need no noise," *Neural Networks*, vol. 119, pp. 200–213, 2019.

[16] D. Probst, M. A. Petrovici, I. Bytschok, J. Bill, D. Pecevski, J. Schemmel, and K. Meier, "Probabilistic inference in discrete spaces can be implemented into networks of lif neurons," *Frontiers in computational neuroscience*, vol. 9, p. 13, 2015.

[17] A. F. Kungl, S. Schmitt, J. Klähn, P. Müller, A. Baumbach, D. Dold, A. Kugele, N. Gürtler *et al.*, "Generative models on accelerated neuromorphic hardware," *arXiv preprint arXiv:1807.02389*, 2018.

[18] G. E. Hinton, T. J. Sejnowski, and D. H. Ackley, *Boltzmann machines: Constraint satisfaction networks that learn*. Carnegie-Mellon University, Department of Computer Science Pittsburgh, 1984.

[19] S. Schmitt, J. Klähn, G. Bellec, A. Grübl, M. Güttler, A. Hartel, S. Hartmann, D. Husmann *et al.*, "Neuromorphic hardware in the loop: Training a deep spiking network on the brainscales wafer-scale system," in *2017 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2017, pp. 2227–2234.

[20] M. A. Petrovici, S. Schmitt, J. Klähn, D. Stöckel, A. Schroeder, G. Bellec, J. Bill, O. Breitwieser *et al.*, "Pattern representation and recognition with accelerated analog neuromorphic systems," in *2017 IEEE International Symposium on Circuits and Systems (ISCAS)*. IEEE, 2017, pp. 1–4.

[21] R. S. Sutton and A. G. Barto, *Reinforcement learning: An introduction*. MIT press, 2018.

[22] N. Frémaux and W. Gerstner, "Neuromodulated spike-timing-dependent plasticity, and theory of three-factor learning rules," *Frontiers in neural circuits*, vol. 9, p. 85, 2016.

[23] T. Wunderlich, A. F. Kungl, E. Müller, A. Hartel, Y. Stradmann, S. A. Aamir, A. Grübl, A. Heimbrecht *et al.*, "Demonstrating advantages of neuromorphic computation: a pilot study," *Frontiers in Neuroscience*, vol. 13, p. 260, 2019.

[24] N. Frémaux, H. Sprekeler, and W. Gerstner, "Functional requirements for reward-modulated spike-timing-dependent plasticity," *Journal of Neuroscience*, vol. 30, no. 40, pp. 13 326–13 337, 2010.

[25] A. J. Holtmaat, J. T. Trachtenberg, L. Wilbrecht, G. M. Shepherd, X. Zhang, G. W. Knott, and K. Svoboda, "Transient and persistent dendritic spines in the neocortex in vivo," *Neuron*, vol. 45, no. 2, pp. 279–291, 2005.

[26] Y. Loewenstein, A. Kuras, and S. Rumpel, "Multiplicative dynamics underlie the emergence of the log-normal distribution of spine sizes in the neocortex in vivo," *Journal of Neuroscience*, vol. 31, no. 26, pp. 9481–9488, 2011.

[27] D. Kappel, S. Habenschuss, R. Legenstein, and W. Maass, "Synaptic sampling: A bayesian approach to neural network plasticity and rewiring," in *Advances in Neural Information Processing Systems*, 2015, pp. 370–378.

[28] A. Knoblauch and F. T. Sommer, "Structural plasticity, effectual connectivity, and memory in cortex," *Frontiers in neuroanatomy*, vol. 10, p. 63, 2016.

[29] S. Billaudelle, B. Cramer, M. A. Petrovici, K. Schreiber, D. Kappel, J. Schemmel, and K. Meier, "Structural plasticity on an accelerated analog neuromorphic hardware system," *arXiv preprint arXiv:1912.12047*, 2019.

[30] C. Liu, G. Bellec, B. Vogginger, D. Kappel, J. Partzsch, F. Neumärker, S. Höppner, W. Maass *et al.*, "Memory-efficient deep learning on a spinnaker 2 prototype," *Frontiers in neuroscience*, vol. 12, 2018.

[31] R. A. Fisher, "The use of multiple measurements in taxonomic problems," *Annals of eugenics*, vol. 7, no. 2, pp. 179–188, 1936.

[32] A.-S. Chiang, C.-Y. Lin, C.-C. Chuang, H.-M. Chang, C.-H. Hsieh, C.-W. Yeh, C.-T. Shih, J.-J. Wu *et al.*, "Three-dimensional reconstruction of brain-wide wiring networks in drosophila at single-cell resolution," *Current Biology*, vol. 21, no. 1, pp. 1–11, 2011.

[33] S. Takemura, A. Bharioke, Z. Lu, A. Nern, S. Vitaladevuni, P. K. Rivlin, W. T. Katz, D. J. Olbris *et al.*, "A visual motion detection circuit suggested by drosophila connectomics," *Nature*, vol. 500, no. 7461, p. 175, 2013.

[34] S.-y. Takemura, Y. Aso, T. Hige, A. Wong, Z. Lu, C. S. Xu, P. K. Rivlin, H. Hess *et al.*, "A connectome of a learning and memory center in the adult drosophila brain," *Elife*, vol. 6, p. e26975, 2017.

[35] K. Neuser, T. Triphan, M. Mronz, B. Poeck, and R. Strauss, "Analysis of a spatial orientation memory in drosophila," *Nature*, vol. 453, no. 7199, p. 1244, 2008.

[36] T. Stone, B. Webb, A. Adden, N. B. Weddig, A. Honkanen, R. Templin, W. Wcislo, L. Scimeca *et al.*, "An anatomically constrained model for path integration in the bee brain," *Current Biology*, vol. 27, no. 20, pp. 3069–3085, 2017.

[37] N. W. Gouwens and R. I. Wilson, "Signal propagation in drosophila central neurons," *Journal of Neuroscience*, vol. 29, no. 19, pp. 6239–6249, 2009.

[38] L. C. Moreaux and G. Laurent, "Estimating firing rates from calcium signals in locust projection neurons in vivo," *Frontiers in neural circuits*, vol. 1, p. 2, 2007.

[39] S. B. Furber, F. Galluppi, S. Temple, and L. A. Plana, "The spinnaker project," *Proceedings of the IEEE*, vol. 102, no. 5, pp. 652–665, 2014.

[40] M. Davies, N. Srinivasa, T.-H. Lin, G. Chinya, Y. Cao, S. H. Choday, G. Dimou, P. Joshi *et al.*, "Loihi: A neuromorphic manycore processor with on-chip learning," *IEEE Micro*, vol. 38, no. 1, pp. 82–99, 2018.

[41] J. Schemmel, D. Brüderle, A. Grübl, M. Hock, K. Meier, and S. Millner, "A wafer-scale neuromorphic hardware system for large-scale neural modeling," in *Proceedings of 2010 IEEE International Symposium on Circuits and Systems*. IEEE, 2010, pp. 1947–1950.

[42] M. A. Petrovici, B. Vogginger, P. Müller, O. Breitwieser, M. Lundqvist, L. Muller, M. Ehrlich, A. Destexhe *et al.*, "Characterization and compensation of network-level anomalies in mixed-signal neuromorphic modeling platforms," *PloS one*, vol. 9, no. 10, p. e108590, 2014.