An error-propagation spiking neural network compatible with neuromorphic processors

Matteo Cartiglia, Germain Haessig and Giacomo Indiveri
Institute of Neuroinformatics
University of Zurich and ETH Zurich, Zurich, Switzerland
Email: [camatteo,germain,giacomo]@ini.uzh.ch

***Abstract*—Spiking neural networks have shown great promise for the design of low-power sensory-processing and edge-computing hardware platforms. However, implementing on-chip learning algorithms on such architectures is still an open challenge, especially for multi-layer networks that rely on the back-propagation algorithm. In this paper, we present a spike-based learning method that approximates back-propagation using local weight update mechanisms and which is compatible with mixed-signal analog/digital neuromorphic circuits. We introduce a network architecture that enables synaptic weight update mechanisms to back-propagate error signals across layers and present a network that can be trained to distinguish between two spike-based patterns that have identical mean firing rates, but different spike-timings. This work represents a first step towards the design of ultra-low power mixed-signal neuromorphic processing systems with on-chip learning circuits that can be trained to recognize different spatio-temporal patterns of spiking activity (e.g. produced by event-based vision or auditory sensors).**

I. Introduction

Spiking neural network (SNN) models are highly efficient at processing sensory signals while minimizing their memory and power-consumption resources. For this reason, they have established themselves as a valuable alternative to traditional deep learning methods for applications that require ultra-low power edge-computing capabilities. These models, compared to standard Artificial Neural Networks (ANNs), take one step closer to brain-inspired processing by using an event-driven processing mode: their leaky Integrate-and-Fire (I&F) neurons transmit information only when there is sufficient input data to reach their spiking threshold. As the I&F neurons are implemented with passive current-mode sub-threshold circuits [1], this data-driven computation mode only burns power when there are signals to process. However, training neural networks implemented with these circuits remains a challenging problem due to their intrinsic spiking non-linearity and to the requirement of local learning rules. To this end, adapting the original back-propagation algorithm [2] to work with spiking neural networks has drawn substantial attention [3]–[5]. So far, however, these attempts have been done for software simulations of spiking neurons and the underlying incongruity between the proposed training mechanism and the limitations of the (noisy, imprecise and low-resolution) mixed-signal analog/digital silicon neurons remains.

Dedicated neuromorphic spiking neural network platforms able to implement learning mechanism have indeed been proposed in recent years [6]–[9]. However, such architectures have been designed with pure digital circuits, or with digital co-processors in the loop. They have not been optimized for power consumption, hence rendering the use of such networks less efficient for low power edge computing applications.

The contribution of this paper, is to investigate the feasibility of creating a dedicated mixed-signal learning circuit that implements a spiking version of a recently proposed cortical model [10] which approximates the back-propagation algorithm through the propagation of local error signals at the network level. We show that this network is able to correctly learn to distinguish different temporal input patterns, relying solely on feedback propagation of a teacher stimulus and local weight updates mechanism, compatible with mixed-signal neuromorphic circuits [1]. The main challenge in this work was to find a spiking neural network architecture that is able to approximate the back-propagation algorithm and at the same time is compatible with mixed-signal learning circuits.

II. Network topology

Inspired by the functionality, connectivity, and diversity of cell types found in the neocortex, previous work [10] has proposed a cortical model of learning and neural computation comprising of a population of rate-based multi-compartment excitatory neurons and inhibitory interneurons. Multi-compartment neurons aim to approximate dendritic processes more accurately by having separate compartments, which interact with each other. Figure 1 shows a simplified representation of such a model: the neuron is schematized as a three-compartment ((A) Apical, (B) Basal and (S) Somatic compartment) pyramidal neuron (P) which integrates sensory bottom-up information and a top-down teaching signal. Pyramidal neurons are driven by sensory inputs, while interneurons (I) are laterally driven by the pyramidal neurons. The apical compartment (A) of the pyramidal neurons receive feedback from higher-order areas as well as from the interneurons.

All bottom-up connections, as well as lateral recurrent connections, (see Figure 1) are subject to plasticity, while the synapses that modulate the top-down teaching signals are fixed and conductance-based (see Figure 1, dotted connections, Eq. 4 and [11]). Plasticity is driven by the requirement of the apical compartment to match the top-down teaching signal coming from the layer above with the lateral connectivity coming from the interneurons. This requires the inhibitory interneuron to match the activity of the above layer. As such

978-1-7281-4922-620$31.00 ©2020 IEEE

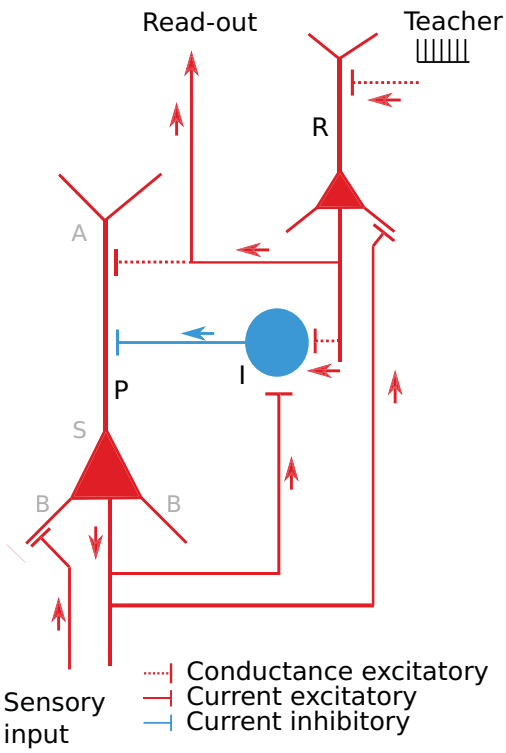

Fig. 1: Network connectivity between (P)yramidal, (I)nter and (R)eadout neurons. Sensory input is propagated bottom-up via the pyramidal neuron; the teacher signal is propagated top-down via the readout neuron. Recurrent connections between pyramidal neurons and interneurons are set-up to minimize the prediction error in the apical compartment of the pyramidal neuron. Dotted connection are conductance-base excitatory synapses, as described in Section III-A

the hidden layer has a number of interneurons equal to the number of neurons in the layer above.

During learning, interneurons learn to replicate the spiking activity of specific neurons in the above layer, such as to cancel the top-down teacher signal in the apical compartment of the pyramidal neuron [12]. Similarly, the firing rate of the readout neuron (R) will tend to the teacher signal. The weights of the network converge so that the firing rate of the readout neuron will tend to the teacher signal, even when the teacher signal is turned off.

III. Constraints and Models for the Hardware Implementation

A. Equations

In the following, we adapted the model to use equations and plasticity mechanisms that can be directly implemented with sub-threshold neuromorphic circuits as presented in [1], and that are implemented on the DYNAP-SE chip [13].

In the theoretical model, the voltage V^k_{Comp} of individual compartments is approximated by low-pass filtering the incoming current I_k of the k–th compartment (Eq. 1). For compatibility with the current-mode neuromorphic circuits, the membrane voltage, is represented as a current I_S, as defined in Eq. 2. This membrane *current* will trigger an action potential when crossing a predefined threshold.

$$I^k_{Comp} = -\sum_i \frac{dI_i}{dt} \tag{1}$$

$$\frac{dI_S}{dt} = \frac{1}{\tau}\sum_k \frac{\alpha_k}{\alpha_k + l} I^k_{Comp} - lI_S + \sigma^2 \tag{2}$$

where τ is the membrane time constant, α_k the coupling factor of the k–th compartment, l the leak and σ^2 an additional noise term (see Section III-B).

Bottom-up weight updates (Figure 1, arrowed connections) are modulated by the learning rule in Eq. 3. The postscripts S and B represent different compartments of the neuron (such as the somatic and basal compartments), but the same learning rule takes place in the (A) apical compartment of the pyramidal neuron between the top-down projection and the lateral inhibitory connection. The hysteresis term (Θ) of the equation gives rise to a stop-learning region which enables life long learning [14].

Bottom-up connections follow the following learning rule :

$$\Delta w = \begin{cases} \eta(I_S - I_B) - \Theta & \text{if } I_S - I_B > \Theta \\ \eta(I_S - I_B) + \Theta & \text{if } I_S - I_B < -\Theta \\ 0 & \text{else} \end{cases} \tag{3}$$

where Δw is the incremental weight update to be applied, being a function of the difference between somatic and basal membrane currents I_S and I_B, η a learning update rate and Θ an hysteresis term.

Top-down conductance-based connections (Figure 1, dotted connections) are described by Eq. 4.

$$I^A_{Comp} = I_{syn}\alpha(I_S - E_{rev}) \tag{4}$$

Where I^A_{Comp} is the apical compartment current injected into the somatic compartment, I_S is the somatic membrane current and E_{rev} is the reversal potential. Finally, α is a scaling factor and I_{syn} is the post-synaptic current.

B. Parameters mismatch

This work was carried out using the Python Teili software library [15] for the Brian spiking neural network simulator [16] which reduces the gap between software simulation and hardware emulation by directly simulating the properties of the mixed-signals CMOS neuron circuits. In order to model the device-mismatch effects due to the fabrication process of circuits, all the variables in the neuron and synapse network are subject to a 20% random variability, centered around their nominal value, which is in the range of the expected variations [17], [18].

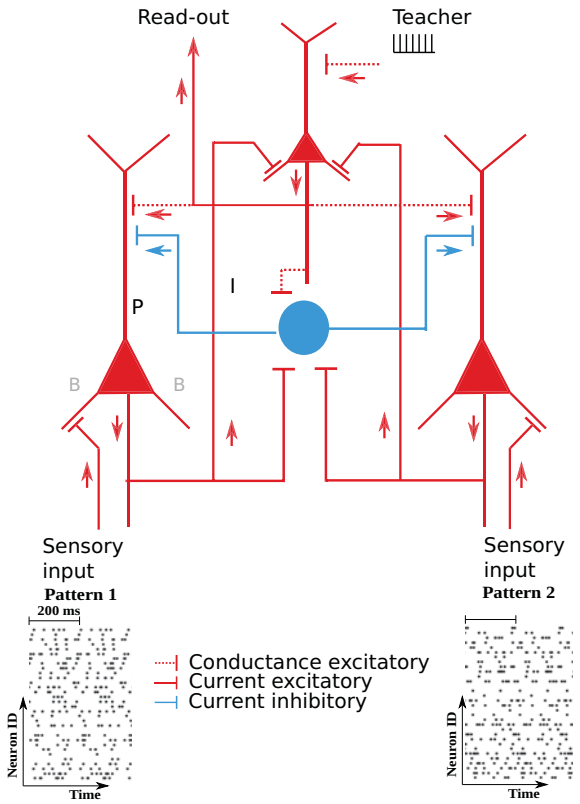

Fig. 2: Architecture of the network. The two input patterns have identical statistics: 128 spikes for a 200ms pattern, distributed on 32 input neurons.

C. Learning Circuits

The learning rules described in section III-A have already been implemented in previously proposed neuromorphic hardware [19]. Conductance based synapses, such as the top-down connections in this work, have been described and tested thoroughly [11]. Using the NMDA block of single neurons, these equations are implementable on mixed-signal neuromorphic processors and have been used in spiking dendritic prediction algorithms [20]. The hardware implementation of the bottom-up connections have been implemented in recent learning circuits [21] and can be extended to exploit the stochasticity of memristive devices.

IV. Results

Following the rate-based model previously proposed [10], we validate the spiking network in a spatio-temporal pattern recognition task: two distinct spatio-temporal patterns, with same mean firing rates, are presented to the network, which was trained to respond only to one of the two. A network comprising two pyramidal neurons and one interneuron is able to learn to detect a 32 channel input sensory pattern. Figure 2 describes the proposed network. A synthetic randomly generated periodic spiking pattern was presented together with a 900 Hz teaching signal. The teacher signal was presented for

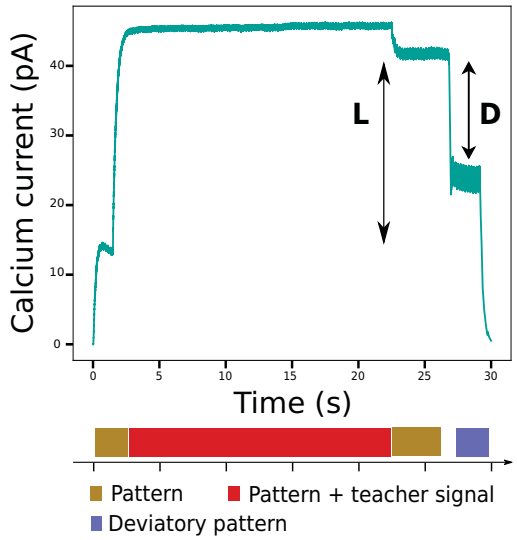

Fig. 3: Results from the pattern recognition task. The plot shows the calcium current which is approximated by the low-passed (200ms time constant) spiking activity of the readout neuron. Spikes are not shown due to their oscillatory dynamics. The amplitude L represents the learning process: the difference in activity of the output neuron between before and after the presentation of teaching signal and input pattern. The amplitude D represents the difference in activity between a known pattern (Pattern 1) and a deviatory sensory input (Pattern 2), proving that learning is specific. Optimally, we would expect L and D to be as large as possible. The spiking activity of the output neuron is measured as a current due to hardware constraints as explained in III-A.

an arbitrary long period during which the network learns to associate the higher firing activity due to the teaching signal with an input pattern and nudges the weights in order to sustain this activity even when the teaching signal is turned off. It is then able to differentiate the input pattern (Pattern 1) from a deviating pattern (Pattern 2) with identical statistics.

Figure 3 shows the results of the pattern recognition task: first, the input pattern is presented without the teaching signal, then the teaching signal and the input pattern are presented together. Finally, the teaching signal is removed. The activity of the output neuron has increased by a factor L due to learning process that took place. After a period in which no input is presented, a second, deviatory pattern is presented to the network: the activity of the output neuron is lower by a factor of D. The response of the untrained network to either pattern is comparable, signifying that the learning process is selective to the shown pattern.

Figure 4 shows how the input weights get modified during the training phase. The different distribution of the input weights between before and after learning proves that the network becomes selective for specific inputs that, combined, enable the output neuron to match the desired target.

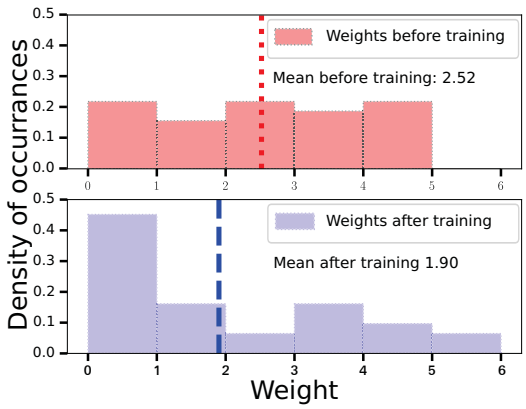

Fig. 4: Input weight distribution before (in red, top) and after (in blue, bottom) learning. The density of occurrences indicates the probability density distribution of the weights. Weights are initialized randomly from a uniform distribution and can evolve freely. The different distributions proves that after learning the network becomes selective to specific inputs.

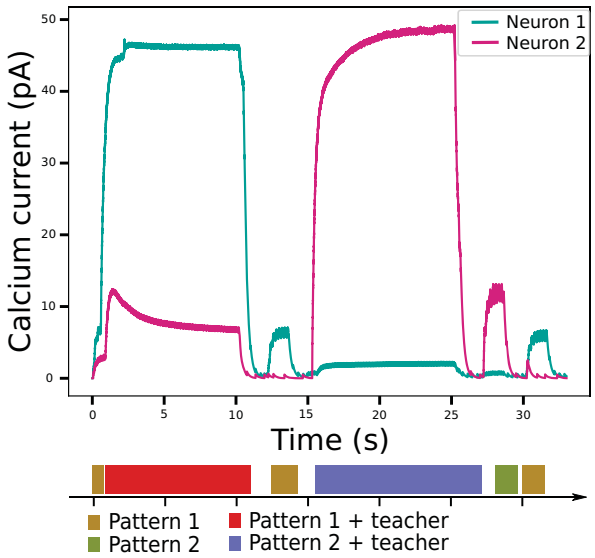

Fig. 5: Results of the pattern discrimination task. The Calcium current plotted in the figure represents the average firing rate (low-pass filtered with a time constant of 200 ms) of the readout neuron. After learning, when the teacher signal is turned off, the network can correctly distinguish the two patterns. During classification, the neuron that encodes the pattern which is not presented is silent.

A. *Pattern discrimination*

Building on the results of the pattern recognition task, described above, we improved the network to be able to correctly classify separate temporal patterns. The hidden layer of the network has two interneurons and 8 pyramidal neurons. The teaching signal, as well as the two input patterns, are kept the same between experiments.

Figure 5 shows the results of the pattern discrimination task. Each sensory pattern is fed in the presence of the teaching signal, specifically only the output neuron that is attributed in classifying the input pattern is stimulated. After having trained the network on both patterns, each pattern is presented in the absence of the teaching signal. The spiking activity of the output neurons, correctly classify the input patterns. In order to optimize the hardware resources at our disposal (limited number of synapses on neuromorphic chips [13]), synaptic connections were created with 50% sparsity constraint: each sensory neuron has a 50% probability of being connected to any neuron in the hidden layer and analogously the neurons in the hidden layer have 50% probability of being connected to any neuron in the readout layer. Weights are randomly initialized with uniform distributions, and can freely evolve. It should be noted that the teacher signal is presented for the same amount of time for both patterns.

V. Conclusion

In this paper, we leverage existing spike-based learning circuits to propose a biologically plausible architecture which, through forward and backward propagation of sensory input and error signals, can successfully classify distinct complex spatio-temporal spike patterns. Such a task, is non-trivial due to the nature of the spatio-temporal patterns. This new architecture relies solely on weight updates triggered by local variables and parameters, thus being suitable for implementation on mixed-signal analog/digital neuromorphic chips. This will enable the construction of low power always-on learning chips that can be applied to edge computing, robotics, and distributed computation. Due to the low-power nature of the hardware the network is implemented for, we expect it to be a good candidate for bio-signals processing [22] and brain-machine interfaces [23]. Further work will include the detailed circuit-level implementation and simulation of the network layout, including transient and Monte-Carlo simulations to fully validate the network's robustness to parameter mismatch and variability.

Acknowledgment

The authors would like to thank Joao Sacramento for fruitful discussions and comments, Ole Richter, Felix Bauer and Melika Payvand for original ideas and insight about the circuit and equations.

This work was partially supported by the Swiss National Science Foundation Sinergia project #CRSII5-18O316 and the ERC Grant "Neu-roAgents" (724295).

REFERENCES

[1] E. Chicca, F. Stefanini, C. Bartolozzi, and G. Indiveri, "Neuromorphic electronic circuits for building autonomous cognitive systems," *Proceedings of the IEEE*, vol. 102, no. 9, pp. 1367–1388, 9 2014.

[2] D. E. Rumelhart, G. E. Hintont, and R. J. Williams, "Learning representations by back-propagating errors," *Nature*, vol. 323, no. 6088, pp. 533–536, 1986.

[3] J. H. Lee, T. Delbruck, and M. Pfeiffer, "Training deep spiking neural networks using backpropagation," *Frontiers in Neuroscience*, vol. 10, p. 508, 2016. [Online]. Available: http://journal.frontiersin.org/article/10.3389/fnins.2016.00508

[4] E. O. Neftci, C. Augustine, S. Paul, and G. Detorakis, "Event-driven random back-propagation: Enabling neuromorphic deep learning machines," *Frontiers in Neuroscience*, vol. 11, p. 324, 2017. [Online]. Available: https://www.frontiersin.org/article/10.3389/fnins.2017.00324

[5] B. Rueckauer, I.-A. Lungu, Y. Hu, M. Pfeiffer, and S.-C. Liu, "Conversion of continuous-valued deep networks to efficient event-driven networks for image classification," *Frontiers in neuroscience*, vol. 11, p. 682, 2017.

[6] J. Schemmel, J. Fieres, and K. Meier, "Wafer-scale integration of analog neural networks," in *Proceedings of the IEEE International Joint Conference on Neural Networks*, 2008.

[7] C. Frenkel, J. Legat, and D. Bol, "A 0.086-mm^2 9.8-pJ/SOP 64k-synapse 256-neuron online-learning digital spiking neuromorphic processor in 28nm CMOS," *CoRR*, vol. abs/1804.07858, 2018. [Online]. Available: http://arxiv.org/abs/1804.07858

[8] P. A. Merolla, J. V. Arthur, R. Alvarez-Icaza, A. S. Cassidy, J. Sawada, F. Akopyan, B. L. Jackson, N. Imam, C. Guo, Y. Nakamura, B. Brezzo, I. Vo, S. K. Esser, R. Appuswamy, B. Taba, A. Amir, M. D. Flickner, W. P. Risk, R. Manohar, and D. S. Modha, "A million spiking-neuron integrated circuit with a scalable communication network and interface," *Science*, vol. 345, no. 6197, pp. 668–673, Aug 2014.

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

[10] J. Sacramento, R. P. Costa, Y. Bengio, and W. Senn, "Dendritic cortical microcircuits approximate the backpropagation algorithm," in *Advances in Neural Information Processing Systems*, 2018, pp. 8721–8732.

[11] C. Bartolozzi and G. Indiveri, "Synaptic dynamics in analog VLSI," *Neural Computation*, vol. 19, no. 10, pp. 2581–2603, Oct 2007.

[12] R. Urbanczik and W. Senn, "Learning by the dendritic prediction of somatic spiking," *Neuron*, vol. 81, no. 3, pp. 521–528, 2014.

[13] S. Moradi, N. Qiao, F. Stefanini, and G. Indiveri, "A scalable multicore architecture with heterogeneous memory structures for dynamic neuromorphic asynchronous processors (DYNAPs)," *Biomedical Circuits and Systems, IEEE Transactions on*, vol. 12, no. 1, pp. 106–122, Feb. 2018.

[14] M. Payvand and G. Indiveri, "Spike-based plasticity circuits for always-on on-line learning in neuromorphic systems," pp. 1–5, 2019.

[15] M. Milde, A. Renner, R. Krause, A. M. Whatley, S. Solinas, D. Zendrikov, N. Risi, M. Rasetto, K. Burelo, and V. R. C. Leite, "teili: A toolbox for building and testing neural algorithms and computational primitives using spiking neurons," 2018, unreleased software, Institute of Neuroinformatics, University of Zurich and ETH Zurich.

[16] D. Goodman and R. Brette, "Brian: a simulator for spiking neural networks in Python," *Frontiers in Neuroinformatics*, vol. 2, 2008.

[17] T. Serrano-Gotarredona and B. Linares-Barranco, "Systematic width-and-length dependent CMOS transistor mismatch characterization and simulation," *Analog Integrated Circuits and Signal Processing*, vol. 21, no. 3, pp. 271–296, December 1999.

[18] A. Pavasović, A. Andreou, and C. Westgate, "Characterization of subthreshold MOS mismatch in transistors for VLSI systems," *Journal of VLSI Signal Processing*, vol. 8, no. 1, pp. 75–85, July 1994.

[19] N. Qiao, H. Mostafa, F. Corradi, M. Osswald, F. Stefanini, D. Sumislawska, and G. Indiveri, "A reconfigurable on-line learning spiking neuromorphic processor comprising 256 neurons and 128k synapses," *Frontiers in neuroscience*, vol. 9, p. 141, 2015.

[20] D. Sumislawska, N. Qiao, M. Pfeiffer, and G. Indiveri, "Wide dynamic range weights and biologically realistic synaptic dynamics for spike-based learning circuits," in *International Symposium on Circuits and Systems, (ISCAS), 2016*. IEEE, 2016, pp. 2491–2494.

[21] M. Payvand, L. K. Muller, and G. Indiveri, "Event-based circuits for controlling stochastic learning with memristive devices in neuromorphic architectures," in *Circuits and Systems (ISCAS), 2018 IEEE International Symposium on*. IEEE, 2018, pp. 1–5.

[22] E. Donati, M. Payvand, N. Risi, R. Krause, K. Burelo, T. Dalgaty, E. Vianello, and G. Indiveri, "Processing EMG signals using reservoir computing on an event-based neuromorphic system," in *Biomedical Circuits and Systems Conference, (BioCAS), 2018*. IEEE, Oct. 2018, pp. 1–4.

[23] F. Corradi and G. Indiveri, "A neuromorphic event-based neural recording system for smart brain-machine-interfaces," *Biomedical Circuits and Systems, IEEE Transactions on*, vol. 9, no. 5, pp. 699–709, 2015.