Memristor Based Neuromorphic Adaptive Resonance Theory for One-Shot Online Learning and Network Intrusion Detection

Md Shahanur Alam†
Dept. of Electrical and Computer Engineering,
University of Dayton, Ohio, USA,
alamm8@udayton.edu

Chris Yakopcic
Dept. of Electrical and Computer Engineering,
University of Dayton, Ohio, USA,
cyakopcic@udayton.edu

Guru Subramanyam
Dept. of Electrical and Computer Engineering,
University of Dayton, Ohio, USA,
gsubramanyam@udayton.edu

Tarek M. Taha
Dept. of Electrical and Computer Engineering,
University of Dayton, Ohio, USA,
tarek.taha@udayton.edu

ABSTRACT

As computer networks become more advanced, the necessity for reliable intrusion detection at extreme efficiency has vastly increased. Thus, in this work we present a one shot learning system capable of on-line learning for network intrusion detection. Adaptive resonance theory is implemented in custom low power memristor-based neuromorphic hardware. The system is capable of discriminating with existing knowledge to learn incrementally. The winner take all circuitry is implemented with a capacitor and CMOS timing circuit that finds the winning neuron and controls the weight update for only the winning neuron. The time required to find a winning neuron was determined to be in the nanosecond range. The performance of the system was evaluated on both previously known and zero-day datasets. The detection accuracy using zero-day packets is 99.97%, and 99.99% for the known attacks. Furthermore, the system was tested using various vigilance parameters and learning rates. The variation of threshold voltage across the capacitor was also investigated to observe the effect on learning and detection accuracy.

CCS CONCEPTS

• Hardware • Emerging technologies • Analysis and design of emerging devices and systems • Emerging architectures

KEYWORDS

ART, Memristor, WTA, Online learning, One-shot fast learning, Network Intrusion

ACM Reference Format:
Md Shahanur Alam, Chris Yakopcic, Guru Subramanyam, and Tarek M. Taha. 2020. Memristor Based Neuromorphic Adaptive Resonance Theory for OneShot Online Learning and Network Intrusion Detection. In *International Conference on Neuromorphic Systems 2020 (ICONS 2020), July 28–30, 2020*

ICONS 2020, July 28–30, 2020, Oak Ridge, TN, USA

ACM ISBN 978-1-4503-8851-1/20/07. . . $15.00
https://doi.org/10.1145/3407197.3407608

Oak Ridge, TN, USA. ACM, New York, NY, USA, 8 pages.
https://doi.org/10.1145/3407197.3407608

1 INTRODUCTION

Internet communication infrastructure is changing and thriving faster than ever before. Millions of devices are connected to the internet through various communication channels, and these devices are generating and sharing information continuously. With the abundance of information flow, the risk of a network breach has also increased [1]. Real-time learning of network packets for intrusion detection is in high demand for an uninterrupted network monitoring system [2]. There are two primary types of network intrusion detection systems available: (1) signature-based intrusion detection systems, and (2) anomaly-based systems [3]. In signature-based detection, any incoming packet is compared with a list of known signatures and identified as either benign or harmful. However, rule-based systems are not very effective once a brand new threat comes into the network [3]. For developing an online, adaptable learning system, the neural network is a feasible solution [2]. Neural networks have the tunability and self-organizing capabilities required to learn and detect a new features present in data [4]. These characteristics are significant when attempting to detect zero-day attacks in real time.

There are two main learning techniques used in neural networks: supervised and unsupervised learning. Supervised learning techniques learn using labeled data and are extremely good at image processing, recognition, and classification [5, 6]. Unlike supervised learning, unsupervised learning techniques do not need any label data to learn patterns or sequences [7]. For online and real-time learning, unsupervised methods are a primary candidate [2]. Fundamental problems when training neural networks include the high power and memory requirements. It is common practice to execute neural networks using Graphics Processing Units (GPUs) that require ~200W of power [8]. Thus, utilizing a GPU to operate neural networks in battery-powered devices such as Internet of Things (IoT) and edge devices is not feasible [9].

The millions of IoTs and edge modules are connected to the internet and are all sending data through different networks in real time. These devices are performing very sophisticated tasks

like automation, industrial processes, human health analysis, and environmental monitoring [10]. The integration of real-world objects with the internet brings network security threats into the realm of our daily activities. Real-time anomaly detection has a high demand in the network security industry. Besides intrusion detection, other critical applications of these types of systems include malware detection, automated fault detection, and medical anomaly detection [11].

To enable online learning and real-time intrusion and anomaly detection in low power devices, nanoscale memristor [12] devices are a suitable candidate. The memristors require very little power, and these devices are extremely efficient when performing multiplication and addition simultaneously in parallel in the analog domain [13-15]. Memristor devices are also able to store information based on an adjustment of their internal resistance [14]. Thus, unlike GPU based architectures, a memristor-based neuromorphic system can be a Non-Von Neumann architecture [15] and does not require an external memory and bus to store and relay data [16]. Neural networks have already been implemented in memristor crossbar circuits for supervised and unsupervised learning for image classification, recognition, and intrusion detection [17-19].

In this study, we are proposing memristor based fast one-shot online learning for network intrusion detection in real-time. We have implemented unsupervised Adaptive Resonance Theory (ART), which is quite stable and does not suffer from catastrophic forgetting [20]. This study presents a full circuit implementation of ART in memristor circuits along with the necessary CMOS peripheral circuits. The ART algorithm is based on the theory of human cognitive information processing [21]. It can be explained as an algorithm of incremental clustering which aims at forming multi-dimensional clusters, automatically discriminating and creating new categories [22]. While learning, ART will initialize with a single output neuron, then increase the output nodes incrementally based on the similarity measure of the incoming packets. After performing the one-pass training, the network achieved greater than 99% testing accuracy. The memristor-based neuromorphic implementation of ART has the potential to provide online learning in real time in IoT and edge devices.

The rest of the paper is organized as follows: Section II describes related works, and Section III presents the dataset used for the experiments. Section IV presents ART fundamentals along with the memristor crossbar implementation, including the winner take all circuits. Section V explains the online training in ART, and Section VI presents the experimental setup. Section VII presents the results and relevant discussion of the outcomes. The paper ends with a brief conclusion in Section VIII.

2 RELATED WORKS

In our earlier work [19], we implemented an AutoEncoder (AE) based online learning system. The system was implemented using two autoencoders for online data processing. One autoencoder contained pretrained knowledge, and the second autoencoder learned in real-time. The drawback with the AE model is catastrophic forgetting when learning a new class [23]. To the best of our knowledge, there is no other work in the literature that discusses unsupervised online learning implemented in memristor based neuromorphic hardware. However, there are a few works on the spiking neural network (SNN) neuromorphic systems which have presented online learning. Recently, the SNN was introduced for online learning for distributed IoT modules [37]. Work in [38] presents a supervised online learning technique, so the system does not emulate real cognitive activity. Many researchers have been working on software learning platforms, but they are not tied to any hardware implementation. Work in [24] uses unsupervised learning via a denoising autoencoder for incremental learning in an evolving environment. More recently, researchers have been focusing on Self-Organizing Map (SOM) techniques for online learning due to their cognitive learning capability [25]. The Extreme Learning Machine (ELM) algorithm is proposed for real-time intrusion detection in [1]. Work in [2] introduced a hierarchical temporal memory (HTM) based unsupervised real-time anomaly detection system proposed for monitoring a video stream. The Winner Take All (WTA) algorithm is proposed for network anomaly detection in [25]. Adaptive Resonance Theory (ART) is a member of the SOM family that is capable of categorizing main brain operations, and it is fast, scalable, and handy for parallel realizations [26]. Many people have investigated the ART model from the algorithmic point of view [20]. Work in [2] studied network intrusion detection with ART and achieved an accuracy of 96% on normal packets and 98% on malicious packets. Work in [28] presents a novel algorithmic scheme to perform the synaptic learning component of ART networks in memristive hardware for binary inputs, but that work did not provide any circuit implementations. In this work, we are presenting a memristor-based ART circuit implementation for unsupervised one-shot online learning on continuous value inputs, providing the ability to perform network intrusion detection for battery-powered devices.

3 NETWORK DATASET

NSL-KDD is the revised version of the KDD Cup'99 dataset that contains samples of network data packets. The dataset contains both training and testing portions, and they consist of 125,973 and 22,544 samples, respectively [29]. Both datasets have normal and malicious network packets. There are 22 different attack types in the training dataset, but there are 39 attack types in the testing dataset. Given this data, an unsupervised intrusion detection system may be more suited to the detection of these new attack types, as they may appear out of place when compared to normal network data. The normal and malicious packets each have 43 attributes with nominal, binary, and numeric values [29]. The nominal attributes are at the 2nd position (protocol/type), the 3rd position (service), the 4th position (flag), and the 42nd position (attack type). The network packets need to undergo some preprocessing steps before they are fed into the network for training and testing. At first, the nominal attributes are replaced with integers.

Then all features are compressed according to min-max normalization to bound each feature to a value within 0 and 1 (including the integer representations of features 2 through 4). The 42nd position represents the attack type, and this feature is

replaced with a 0 or a 1 for normal and malicious packets, respectively. Example packets from the NSL-KDD dataset are shown in Figure in raw and preprocessed version of the same packets.

Figure 1: **Example of network packets from NSL-KDD dataset in its raw and preprocessed form.**

After preprocessing, 90% of the samples in the training data are used for one-shot training of the ART network system, and the remaining 10% were used for testing. This is the dataset used to determine the network's ability to recognize previously known attacks. Alternatively, for the simulation of zero-day learning, the entire NSL-KDD training dataset is used for training, and the entire NSL-KDD testing dataset (which contains 17 new attack types) is used for testing.

4 MEMRISTOR BASED ART

4.1 ART Fundamentals

ART is a type of unsupervised neural network algorithm. It is a fast and stable incremental learning algorithm with a relatively small memory requirement [26]. Fast learning refers to the ability of the synaptic weight vectors to converge to their asymptotic values directly upon each input sample presentation. The ART algorithm has the ability to balance between plasticity and stability, which makes the algorithm more robust when obtaining new knowledge without suffering from catastrophic forgetting [20] of prior learned knowledge. ART can be scalable for large-sized datasets and is capable of processing noisy data [26].

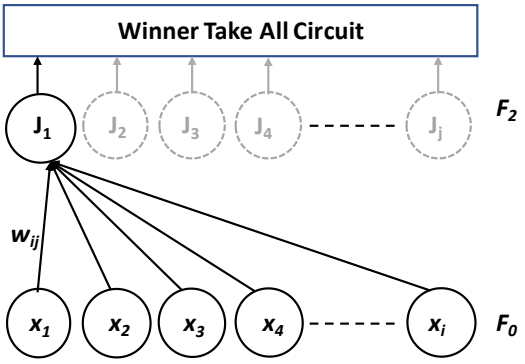

Fig. 2. Block diagram of ART with an active output node J_1 and other possible nodes.

Fig. 2 presents the underlying ART architecture with two layers of neurons. The first layer F_0 is known as the comparison layer, and F_2 is the recognition layer. Once an input is fed to the network, a predefined vigilance parameter identifies the possible candidate(s) from the nodes at the F_2 layer. The winner take all (WTA) method results in choosing only the winning neuron when updating synaptic weights [20]. In the ART network, the F_2 layer is initialized with a single output node J_1 (the circle with the solid line in the F_2 layer). The node with the solid line indicates the initial node and faded nodes indicate the possible nodes to be activated once the active node will discriminate with the incoming sample(s). If the next instances don't match with the first node, then the successive node (or nodes) will be initialized and learn the respective categories. The output of the F_2 layer is represented by equation (1).

$$DP_j = \sum w.x \qquad (1)$$

The activation function of the neurons is described by equation (2), which is known as the choice function. The choice function scales the excitatory signal with the net magnitude of the neural weight. Here, α is a small constant [32].

$$T_j = \frac{DP_j}{\alpha + |w|} \qquad (2)$$

The matching parameter is equal to the scaled value of the dot product for an incoming sample divided by the norm of the input signal, as described as equation (3). The matching function (MF_j) searches for the possible winning neurons compared with a predefined threshold called the vigilance parameter ($0 < \rho < 1$), as in equation (4). If MF_j does not satisfy the vigilance parameter, then the output becomes -1, and ART creates a new neuron with a random weight and assigns the instance belong to the newly created neuron [32].

$$MF_j = \frac{DP_j}{|x|} \qquad (3)$$

$$\text{Output} = \begin{cases} T_j, & if\ MFj > \rho \\ -1, & otherwise \end{cases} \qquad (4)$$

The ART updates the synaptic weight only for the winning neuron, which is determined by the maximum value of T according to the equation (5). The winning neuron j is updated according to equation (6). Here, β is the learning rate and is bounded accordingly: $0 < \beta \leq 1$ [20].

$$w_{winner} = \max(T_j) \qquad (5)$$

$$w_{new} = w_{old}^{j}(1-\beta) + \beta x \qquad (6)$$

4.2 Memristor Implementations

Fig. 3 shows the flowchart of the ART neural network. The vigilance parameter scans the matching function to identify the possible winner neurons, and the WTA finds the winning neuron, which exhibits the maximum choice function. The ART system updates the weight for only the winning neuron until it reaches w_{min}. If there is no winning neuron, then the output is set to -1, and the process deactivates the category and creates a new group and initializes a new node to set the output.

Fig. 4. presents the memristor crossbar for the ART implementation. Each column represents a neuron with a CMOS

control circuit, as shown in the inset of Fig. 4. The detailed control unit is presented in Fig. 5. There is a capacitor connected at the bottom of each neuron. The role of the capacitor is to accumulate the charges for any applied potential.

For this particular implementation of ART, the capacitor starts to accumulate the charges after performing the vigilance test on the neurons. For the vigilance test, we have implemented a comparator as in Fig. 6. The comparator compares the magnitude of the

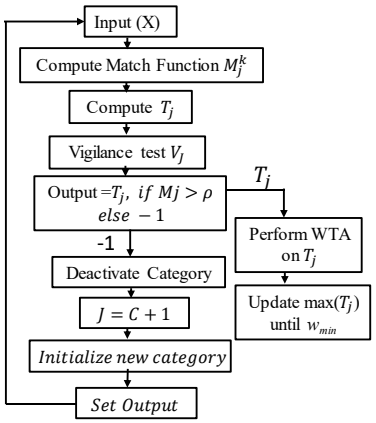

Fig. 3: Flowchart of the ART neural network

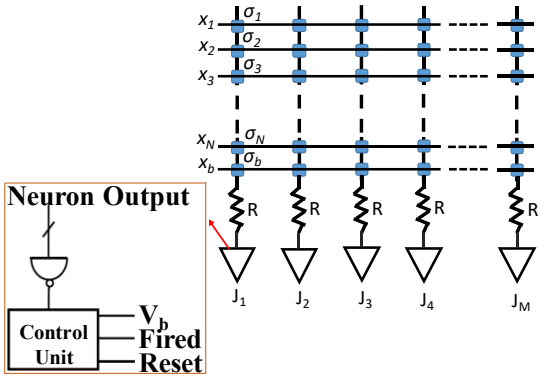

Fig. 4. The memristor crossbar circuit with peripheral CMOS winner take all circuitry to chose the winner neuron

matching function of equation (3) with the vigilance parameter. If $MF_j > \rho$, then the respective capacitor starts charging, otherwise the network switches to a new neuron and initializes it with a random weight.

At the steady-state, the potential across the capacitor is a normalized dot product of the input x and the respective synaptic weight σ as R is a high resistance. For zero bias, the accumulated voltage is described by equation (7). Here, σ_b is a constant value and represents α in equation (2). Equation (2) is analogous to equation (7) which describes the calculation of V_c for an input network packet with 41 features.

$$V_c = x_1\sigma_1 + x_2\sigma_2 + x_2\sigma_2 + \cdots + x_{41}\sigma_{41} + 0.\sigma_b / \sum_{k=1}^{41}\sigma_k + \sigma_b \quad (7)$$

4.3 WTA Circuit Operation

Fig. 5 shows a single neuron with an op-amp comparator and a capacitor for charge accumulation. The capacitor starts to accumulate if the matching function satisfies the vigilance parameter according to equation (4). After turning ON the switches of candidate neurons, the capacitors begin to accumulate. The charging rate will be faster for the neuron with

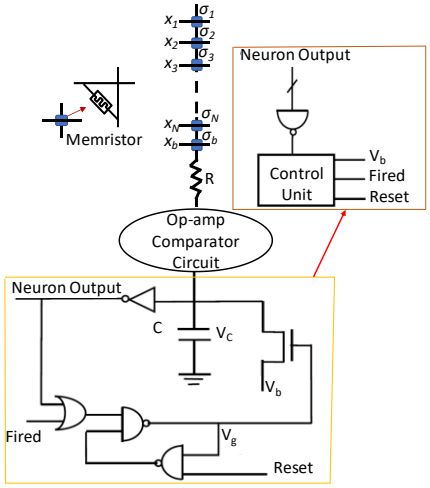

Fig. 5. Single neuron crossbar with necessary winner take all circuitry associated with the neuron for selecting the category.

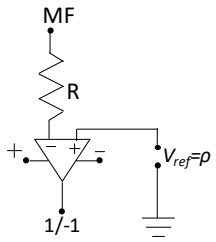

Fig. 6. Op-amp comparator circuit for choosing the neuron those satisfy the vigilance parameter

the highest column voltage according to the equation (7). The charging time is determined from the RC relationship as described in equation (8). Here, R is a high resistance, C is the capacitance of the capacitor, and V_c is the voltage drop across the capacitor. The winning neuron is decided as the first charge accumulating column capacitor voltage crosses the threshold voltage V_i.

$$t_c = RC[log\left(V_c / V_c - V_i\right)] \quad (8)$$

A CMOS inverter is used to detect the moment when the voltage drop across the capacitor passes V_i. The basic operation of this digital circuit is described in our previous work [33]. The average time required to find a winning neuron is few nanoseconds. Once the winning neuron crosses the threshold voltage, V_g switches from low to high, which indicates the individual neuron has fired, and the neuron updates the weight. It is crucial to restrict other neurons from firing after the winning neuron fires. The output of the NAND operation flips high if any neuron output switches from high to low. This neuron firing signal is perpetual; thus, no other neuron can switch V_g to high, and the updating of any other neuron can be deterred.

4.4 Weight Update and Timing Signals

Memristor devices behave like a typical resistor below a threshold voltage (V_{th}). Thus, changing the resistance of a memristor requires an excitation above V_{th}. The device modeled for this study has a threshold voltage of 1.3 V [34].

The weight update is performed according to equation (6). From this equation, we can see that the system needs to read the existing conductance state of the device. To perform the reading of the memristor device and weight update, we have adopted the technique described in [35]. Fig. 7 shows the reading and writing circuit for the memristor devices.

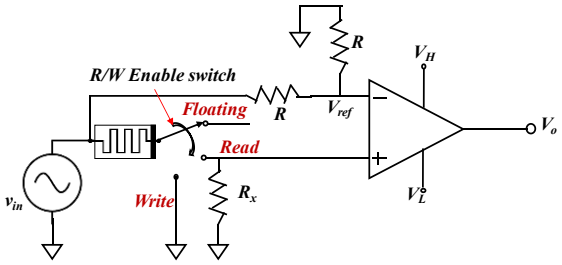

Fig. 7. Memristor reading and writing circuit

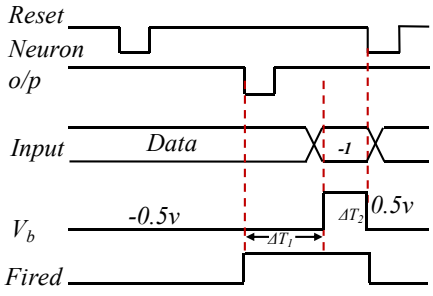

Fig. 8. Timing of the signal during the training period.

In Fig. 8, the timing and amplitude of the signals during the training period are described. We can consider that at a particular time *t*, the *jth* neuron is fired. Just after this firing, V_g of the *jth* neuron flips from low to high, which turns on the NMOS transistor and the capacitor discharges. Ultimately, the V_b becomes -0.5 V on the fired neuron, and the memristors that have high inputs have a voltage across them that exceeds the threshold voltage. Therefore, the conductance of these memristors increases until time $t + \Delta t_1$.

Then, V_b switches to 0.5 V, and each input with a -1 V potential brings the voltage drop across the memristors to -1.5 V. Consequently, the conductance of the memristor decreases. This process continues until $t + \Delta t_1 + \Delta t_2$, where Δt_2 is the learning rate. The reset pulse is applied with V_b = -0.5 V and *Fired* = 0 V. The next training sample then enters the network, and the cycle repeats. This circuit continues to reduce the weight of the appropriate neurons until a minimum conductance value is reached (based on internal memristor properties). Also, the training phase needs to implement a digital counter to count the maximum number of neurons to limit the crossbar size in practical implementation. In this study, the output nodes were limited to 256.

5 ONE-SHOT ONLINE TRAINING OF ART

This algorithm allows for a new output node once it finds an unknown input, or anomaly in the network according to the vigilance parameter [32]. The magnitude of the vigilance parameter plays a vital role in the initialization of a new node. In this experiment, the network packets are presented to the ART network randomly (without any label) only once, and the same packets are not presented multiple times. Despite running multiple epochs or learning cycles, the system utilizes one pass fast learning [36,38]. The model performs clustering-based anomaly detection by focusing on the point anomalies where an individual data instance can be considered as anomalous with respect to the rest of the data [36]. A profile of an incoming packet is assigned to a neuron, then deviation from this profile is regarded as anomalous to the neuron, and the system allows for the creation of a new neuron for this datatype.

After performing the one pass online training, the model was evaluated with two different testing datasets. To implement the ART network in memristor hardware, we have imposed a constraint on making new nodes as the hardware is confined to maximum node number, and cannot make nodes in the same way the human brain assigns neurons for learning a piece of information [37]. For this study, we have constrained the maximum number of nodes to 256, which can be counted by implementing an 8-bit digital counter.

6 EXPERIMENTAL SETUP

The memristor crossbar circuits were simulated in SPICE, which facilitates evaluation of the memristor grid accurately by considering parasitics such as crossbar sneak-paths and wire resistances. The memristor was simulated with an accurate model that was published in [34]. For this study, the off and on state resistances were set to $500k\Omega$ and $100k\Omega$, respectively. The full resistance range of the device can be switched in 100 ns by applying 1.5 V across the device. The values of σ_b, R_h, and C in Fig. 5 are 10 μS, 1 MΩ, and 100 fF, respectively. Once the SPICE circuits were verified, the system was scaled up in MATLAB to facilitate training on a large dataset. We implemented the full ART circuit in MATLAB based on these parameters where the crossbar devices

are able to learn, starting at a random weight value. The threshold voltage of the capacitor, vigilance parameter, and learning rate were varied to study the impact this has on performance.

7 RESULT AND DISCUSSION

We executed the proposed ART based unsupervised one-shot online learning and real-time intrusion detection system on the NSL-KDD dataset. The ART algorithm is implemented in traditional software and the simulated memristor neuromorphic hardware. First, 90% of the training samples were used to train the network online. Training did not utilize any batched learning, instead a single shot learning process was used. Then the 10% of the training packets were used for validation of known samples. The original NSL-KDD test set was used as a testing set, which contains zero-day packets, as this data contains attack types not previously observed during training.

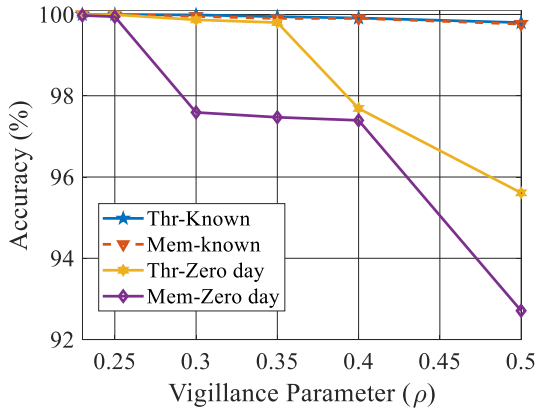

Fig. 9. Accuracy of theoretical ART and memristor-based ART for various vigilance parameters, V_c=0.1V, $\beta = 0.8$

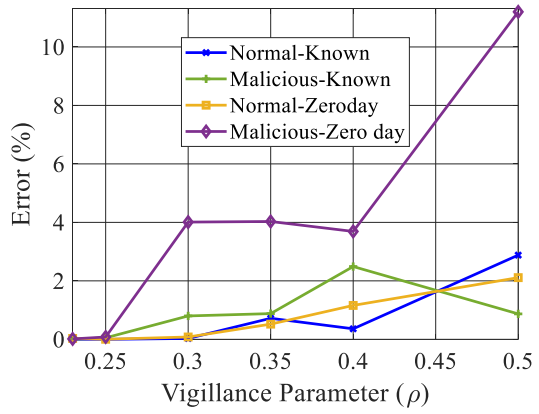

Fig. 10. Percent of error detection Vs. vigilance parameter of memristor ART, V_c=0.1V, $\beta = 0.8$

The vigilance parameter plays a vital role in node selection for an incoming packet. Fig. 9 shows the detection accuracy versus vigilance parameter for the network packets in the theoretical and neuromorphic ART system implemented in the memristor crossbar. The memristor crossbar based system achieved a 99.97% detection accuracy. The baseline theoretical model was implemented in software, and the detection accuracy was 99.99%, which is higher than that of [29]. Both studies were performed with the same initial conditions using a vigilance parameter of 0.23 and learning rate 0.8.The performance of unsupervised ART in memristors also outperformed the memristive autoencoder in [19]. The vigilance parameter (ρ) is varied from 0.23 to 0.5. The learning rate and the threshold voltage of the capacitor in Fig.5 were kept constant at 0.8 and 0.1 V, respectively.

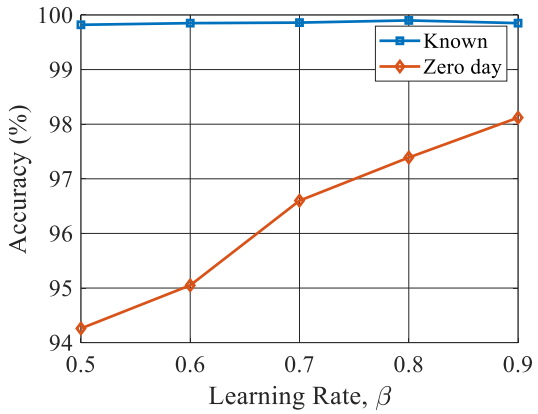

Fig. 11. Accuracy of memristor ART for various learning rate, V_C=0.1V, $\rho = 0.23$

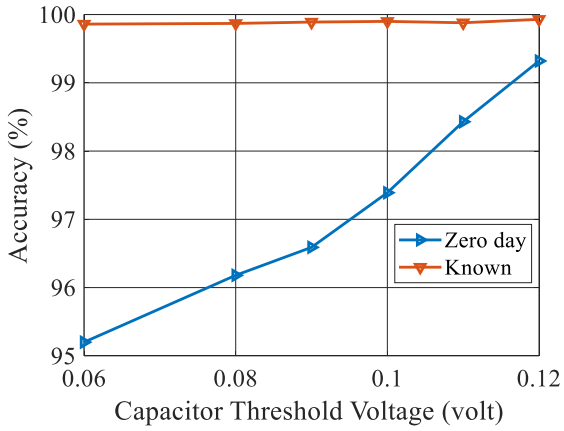

Fig. 12. Accuracy Vs. threshold voltage of capacitor used for triggering the winning neuron, ρ=0.23, $\beta = 0.8$

We can see that at lower values of ρ, the accuracy is higher. This is because at higher ρ values, the system creates more output nodes. Thus, it is more likely that the system selects the wrong node for an incoming sample. The theoretical accuracy for known and zero-day attacks is almost the same at lower vigilance parameters, but error increases at higher vigilance values. Fig. 10 shows the percentage of error in the identification of normal and malicious packets in memristor-based ART. The error is lower for

a vigilance parameter 0.23 and increases at higher values, which is inversely proportional to the accuracy described in Fig. 9. The effect of the learning rate in memristor-based ART is studied and presented in Fig. 11. The accuracy changes a little for known data, but in the case of zero-day packets, the detection accuracy increases as the learning rate increases.

The charging capacitor within each neuron circuit starts to accumulate charge when the possible candidate neurons connected to the WTA circuit are identified after vigilance testing. The capacitor continues charging until reaching the threshold voltage V_i. The neuron with the maximum value choice function will charge the capacitor faster and reach the firing threshold first, and the WTA circuit will fire. The role of threshold voltage was studied and it was found that at higher voltage, the accuracy increases (see Fig. 12). However, maximum ranges for voltage drop for an incoming network packet must be monitored. The maximum threshold voltage (V_i) needs to be lower than the maximum value of T_j. Otherwise, the capacitor charging time (see Equation 8) will become imaginary, which is not pragmatic. The corresponding error values are presented in Fig. 13. The false detection rate decreases as voltage increases, reaching a minimum at the threshold voltage of 0.12V.

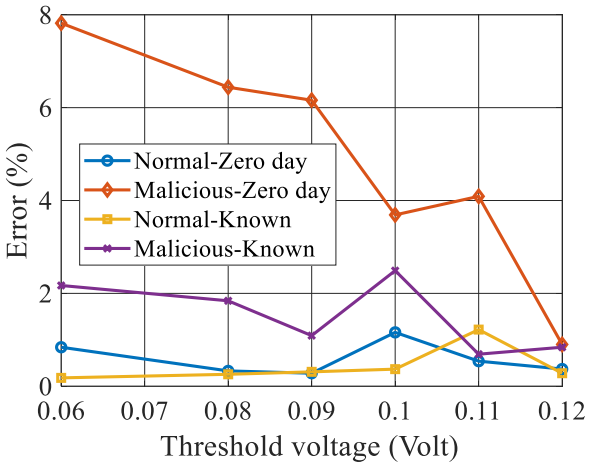

Fig. 13. Percent of Error Vs. threshold voltage of the capacitor, ρ=0.23, $\beta = 0.8$

8 CONCLUSION

An unsupervised, fast one-shot online learning and real-time intrusion detection system was presented based on an ART neural network. The ART system was implemented in a memristor-based neuromorphic system, as well as traditional software. The memristor crossbar system exhibits the exact functionalities of the software-based model. The winner take all, and weight update control circuits were designed to be part of the CMOS peripheral circuitry. A charging capacitor was used to determine the winning neuron by introducing a threshold. The computation was performed in two phases. The training was conducted using online one shot learning and then was tested using both known attack types as well as zero-day attacks. The detection accuracy for the known datatypes was found to be 99.99%, and for the zero-day case, an accuracy of 99.97% was observed. In the future, we plan to investigate the power, energy, and timing of this system to provide a qualitative comparison to alternative architectures.

ACKNOWLEDGMENTS

The work was supported through the National Science Foundation under grants 1718633.

REFERENCES

[1] Setareh Roshana, Yoan Michel, Anton Akusokd and Amaury Lendasse. 2018. *Adaptive and online network intrusion detection system using clustering and Extreme Learning Machines.* Journal of the Franklin Institute Vol. 355, 1752–1779, Iss. 4, (March 2018). DOI: https://doi.org/10.1016/j.jfranklin.2017.06.006.

[2] Sabutai Ahmad, Alexander Lavina, Scott Purdy and Zuha Agha. 2017. *Unsupervised real-time anomaly detection for streaming data.* 262, 134-147, Neurocomputing(2017).DOI:https://doi.org/10.1016/j.neucom.2017.04.070

[3] Matilda Rhode, Pete Burnap and Kevin Jones. 2017. *Early-stage malware prediction using recurrent neural networks.* computers & security, 578-594, 77 (August 2017). DOI: https://doi.org/10.1016/j.cose.2018.05.010.

[4] Doyen Sahoo, Quang Pham, Jing Lu and Steven C. H. Hoi. 2018. *Online Deep Learning: Learning Deep Neural Networks on the Fly.* Twenty-Seventh International Joint Conference on Artificial Intelligence, Pages 2660-2666, Stockholm, (2018). DOI: https://doi.org/10.24963/ijcai.2018/369.

[5] Japkowicz Nathalie. 2018. Supervised Versus Unsupervised Binary-Learning by Feedforward Neural Networks. Machine Learning. 42**,** 97–122 (Jan. 2001). DOI: https://doi.org/10.1023/A:1007660820062.

[6] Gangadhar Shobha and Shanta Rangaswamy. 2018. *Computational Analysis and Understanding of Natural Languages: Principles, Methods, and Applications.* Handbook of Statistics (1 st Ed.). 8, 38, 197-228, 2018. North Holand, Amsterdam, Netherlands.

[7] Stefano Zanero and Sergio M. Savaresi. 2004. *Unsupervised learning techniques for an intrusion detection system.*ACM Symposium on Applied Computing. Nicosia, Cyprus, (March 2004). DOI: https://doi.org/10.1145/967900.967988.

[8] Jacob Ridley. 2019. PCGames^{N}. Retrieved from https://www.pcgamesn.com/nvidia/12nm-vs-amd-7nm-gpu-efficiency-incomparable.

[9] Md Zahangir Alom and Tarek M. Taha. 2017. *Network Intrusion Detection for Cyber Security on Neuromorphic Computing System.*International Joint Conference on Neural Networks (IJCNN), 14-19 May 2017. Anchorage, AK, USA. DOI: https://doi.org/10.1109/IJCNN.2017.7966339.

[10] Elrawy M. Faisal, Awad A. Ismail and Hamed Hesham. 2018. Intrusion detection systems for IoT-based smart environments: a survey. *J Cloud Comp* **7,** 21 (2018). DOI: https://doi.org/10.1186/s13677-018-0123-6.

[11] Dromard Juliette, Roudiere Gilles and Owezarski Philippe. 2015. *Unsupervised Network Anomaly Detection in Real-Time on Big Data.* In: Morzy T., Valduriez P., Bellatreche L. (eds) New Trends in Databases and Information Systems. ADBIS 2015. Communications in Computer and Information Science, vol 539. Springer, Cham.

[12] Leon O. Chua. *1971. Memristor—The Missing Circuit Element.* IEEE Transactions on Circuit Theory. 18, 5, 507–519 (1971). DOI: https://doi.org/10.1109/TCT.1971.1083337.

[13] Chris Yakopcic and Tarek M. Taha. 2018. *Analysis and Design of Memristor Crossbar Based Neuromorphic Intrusion Detection Hardware.* IEEE/INNS International Joint Conference on Neural Networks (IJCNN). 1-7, Rio de Janeiro, Brazil, (July 2018). DOI: https://doi.org/10.1109/IJCNN.2018.8489252.

[14] Raqibul Hasan and Tarek M. Taha. 2018. *Memristor Crossbar Based Low Power Intrusion Detection Systems.* 17th Int'l Conf. on Computer and Information Technology. Dhaka, Bangladesh, 22-23 December 2014. DOI: https://doi.org/10.1109/IJCNN.2018.8489252.

[15] Geoffrey W. Burr. 2016. Accelerating large-scale neural networks with analog non-volatile memory. *1st International Workshop on Memristive Devices for Neuronal Systems.* Kiel, Germany, (September 2016). DOI: https://doi.org/10.1109/IEDM.2015.7409625.

[16] Living Xu, Lin Bao, Teng Zhang, Ke Yang, Yimao Cai, Yuchao Yang, and Ru Huang. 2018. Nonvolatile memristor as a new platform for non-von Neumann computing. 14 th IEEE International Conference on Solid-State and Integrated Circuit Technology (ICSIT). Qingdao, China, (31 Oct. -3 nov. 2018). DOI: https://doi.org/10.1109/ICSICT.2018.8565029.

[17] Chris Yakopcic, M. Zahanir Alom and Taek M. Taha. 2016. *Memristor Crossbar Deep Network Implementation Based on a Convolutional Neural Network.*IEEE/INNS International Joint Conference on Neural Networks (IJCNN), 963-970, Vancouver, BC, (July 2016). DOI: https://doi.org/10.1109/IJCNN.2016.7727302.

[18] Raqibul Hasan and Tarek M. Taha. 2014. *Enabling Back Propagation Training of Memristor Crossbar Neuromorphic Processors.* International Joint Conference on Neural Networks **(***IJCNN***),** Beijing, July 2014. DOI: https://doi.org/10.1109/IJCNN.2014.6889893.

[19] M. Shahanur Alam, B. Rasitha Fernando, Yassine Jaoudi, Chris Yakopcic, Raqibul Hasan, Tarek M. Taha and Guru Subramanyam. 2019. *Memristor Based Autoencoder for Unsupervised Real-Time Network Intrusion and Anomaly Detection.* International Conference on Neuromorphic Computations (ICONS), July 2019, Knoxville. DOI: https://doi.org/10.1145/3354265.3354267.

[20] Leonardo Enzo Brito da Silva, Islam Elnabarawy and Donald C. Wunsch II. 2019. A Survey of Adaptive Resonance Theory Neural Network Models for Engineering Applications. Neural Networks, 120, 167-203, (Dec. 2019). DOI: https://doi.org/10.1016/j.neunet.2019.09.012.

[21] Gail A. Carpenter and Stephen Grossberg. 1987. *A Massively Parallel Architecture for a Self-Organizing Neural Pattern Recognition Machine.* Elsevier, Computer Vision, Graphics and Image Processing, 37, 54-115, (1987). DOI: https://doi.org/10.1016/j.neunet.2019.09.012.

[22] Brahma Anitarani and Panigrahi Suvasini. 2020. Database Intrusion Detection Using Adaptive Resonance Network Theory Model. In: Behera H., Nayak J., Naik B., Pelusi D. (eds) Computational Intelligence in Data Mining. Advances in Intelligent Systems and Computing, vol 990. Springer. DOI: https://doi.org/10.1007/978-981-13-8676-3_22.

[23] Mohammad Rostami, Soheil Koulouri and Praveen K. Pilly. 2019. *Complementary Learning for Overcoming Catastrophic Forgetting Using Experience Replay,*' Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence (IJCAI-19), (May 2019). DOI: https://doi.org/10.24963/ijcai.2019/463.

[24] Ashfahani. 2019. Autonomous Deep Learning: Incremental Learning of Deep Neural Networks for Evolving Data Streams**.** *International Conference on Data Mining Workshops (ICDMW)***,** Beijing, China, (8-11 Nov. 2019). DOI: https://doi.org/10.1109/ICDMW.2019.00023.

[25] Aleksei Kharitonov and Axel Zimmermann. 2019. *Intrusion Detection Using Growing Hierarchical Self-Organizing Maps and Comparison with other Intrusion Detection Techniques.* CPSS, (July 8 2019), Auckland, New Zealand. DOI: https://doi.org/10.1145/3327961.3329531.

[26] Pavlo Tymoshchuk and Serhii Shatnyi. 2019. *Hardware Implementation Design of Parallelized Fuzzy Adaptive Resonance Theory Neural Network.* IEEE XVth International Conference on the Perspective Technologies and Methods in MEMS Design, Polyana, Ukraine, Ukraine, (22-26 May 2019). DOI: https://doi.org/10.1109/MEMSTECH.2019.8817406.

[27] D G Bukhanov and V M Polyakov. 2018. *Detection of network attacks based on adaptive resonance theory.* International Conference Information Technologies in Business and Industry (2018). DOI: https://doi.org/10.1088/1742-6596/1015/4/042007.

[28] Max Versace, Robert T. Kozma and Donald C. Wunsch. 2012. Adaptive Resonance Theory Design in Mixed Memristor-Fuzzy Hardware. Advances in Neuromorphic Memristor Science and Applications, (June 2012). 9, 133-154. DOI: https://doi.org/10.1007/978-94-007-4491-2_9.

[29] UNB. NSL-KDD dataset. Retrieved from https://www.unb.ca/cic/datasets/nsl.html.

[30] Lakhmi C. Jain, Beatrice Lazzerini and Ugur Halici. 2000. *Innovations in ART Neural Networks.* Springer Publisher, ISBN 978-3-7908-1857-4, (2000). Springer-Verlag Berlin and Heidelberg GmbH & Co. KG, Heidelberg, Germany.

[31] B Rasitha Fernando, Raqibul Hasan and Tarek M. Taha. 2018. *Low Power Memristor Crossbar Based Winner Takes All Circuit.* IJCNN, Rio de Janeiro, Brazil, (8-13 July 2018). DOI: https://doi.org/10.1109/IJCNN.2018.8489735.

[32] Chris Yakopcic, Tarek M. Taha, G. Subramanyam, and R. E. Pino. 2013. *Memristor SPICE Model and Crossbar Simulation Based on Devices with Nanosecond Switching Time.* IEEE International Joint Conference on Neural Networks (IJCNN), (4-9 August 2013). DOI: https://doi.org/10.1109/IJCNN.2013.6706773.

[33] Mohamed Elshamy, Hassan Mostafa, Yehya H. Ghallab, and Mohamed Sameh Said. 2014. *A Novel Nondestructive Read/Write Circuit for Memristor-Based Memory Arrays.* IEEE Transaction on VLSI system, 23, 11, 2648 – 2656, 2014. DOI: https://doi.org/10.1109/TVLSI.2014.2377192.

[34] Ngamwtthayanon Nawa, Naruemon Wattanapongsakorn and David W. Coit. 2009. *Investigation of Fuzzy Adaptive Resonance Theory in Network Anomaly Intrusion Detection.*In: Yu W., He H., Zhang N. (eds) Advances in Neural Networks – ISNN 2009. Lecture Notes in Computer Science, vol 5552. Springer, Berlin, Heidelberg.

[35] Sandra Ackerman. 1992. *Discovering the Brain.* 8, 123, National Academy Press, Washington, D.C., USA. </bib>

[36] Chenping Hou and Zhi-Hua Zhou. 2018. *One-Pass Learning with Incremental and Decremental Features.* IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol: 40, Issue: 11, Nov. 2018. DOI: https://doi.org/10.1109/TPAMI.2017.2769047.

[37] Jesus L. Lobo, Javier Del Ser, Albert Bifet and Nikola Kasabov. 2020. *Spiking Neural Networks and online learning: An overview and perspectives.* Neural Networks, 121, 88–100, (2020). DOI: https://doi.org/10.1016/j.neunet.2019.09.004.

[38] G. C. Qiao, S. G. Hu, J. J. Wang, C. M. Zhang, T. P. Chen, N. Ning, Q. Yu and Y. Liu.2019. *A Neuromorphic-Hardware Oriented Bio-Plausible Online-Learning Spiking Neural Network Model. I**EEE Access***, 7, 71730-71740, 2019. DOI: https://doi.org/10.1109/ACCESS.2019.2919163.