TELKOMNIKA, Vol.17, No.3, June 2019, pp.1236~1241
ISSN: 1693-6930, accredited First Grade by Kemenristekdikti, Decree No: 21/E/KPT/2018
DOI: 10.12928/TELKOMNIKA.v17i3.12409

Approximated computing for low power neural networks

Gian Carlo Cardarilli, Luca Di Nunzio*, Rocco Fazzolari, Daniele Giardino, Marco Matta, Mario Patetta, Marco Re, Sergio Spanò
Department of Electronic Engineering, University of Rome Tor Vergata,
Via Del Politecnico 1, Rome, 00133, Italy
*Corresponding author, e-mail: di.nunzio@ing.uniroma2.it

Abstract

This paper investigates about the possibility to reduce power consumption in Neural Network using approximated computing techniques. Authors compare a traditional fixed-point neuron with an approximated neuron composed of approximated multipliers and adder. Experiments show that in the proposed case of study (a wine classifier) the approximated neuron allows to save up to the 43% of the area, a power consumption saving of 35% and an improvement in the maximum clock frequency of 20%.

Keywords*: approximated computing, low power machine learning*

1. Introduction

Machine Learning (ML) plays an important role in several fields as health, computer vision, communications, energy management etc [1-10]. The interest in Machine Learning increased in the last few years. This was possible thanks to the availability of increasing computational power and the introduction of new technologies [11-21]. Also, in embedded systems, there is a growing trend in the use of ML. This is the case for example of Automotive, Security and Surveillance, Smart Home, Health Care, and IoT. For embedded systems, power consumption represents a crucial aspect [22-25]. In fact, these systems are often used under operating conditions where power supply cannot be provided by the electrical grid. In this scenario, reducing the power consumption is one of the most important design goals in order to guarantee a long service life. There are three power dissipation components in CMOS digital circuits [26]:

- Switching Power
- Short-circuit Power
- Static Power

Among these contributions, switching power represents the main one and it is defined in (1), where α is the switching activity, C is the switching capacitance, f is the clock frequency and V_{dd} the supply voltage.

$$P = aCfV_{dd}^2 \quad (1)$$

The second contribution, the short-circuit power, is related to the short-circuit currents flowing through the MOS transistors in the gate at each switching. It is strongly dependent on the parameters present in (1) (switching activity, clock frequency, and supply voltage). Finally, the static power depends on the leakage currents and it is related to the circuit design, the technology, and the supply voltage. ML is characterized by parallel computation and a consequently big area in terms of circuit size. Area impacts negatively on power consumption, with the increasing of the area there is also an increase of all the three power dissipations. There are many techniques to reduce power consumption in digital circuits that can be divided into two main categories, technological solutions, and design solutions. The first ones are based on the using of low power digital libraries, material or devices. The second ones consist of the use of design techniques both at layout level (for example power gating and/or power gating) both at RTL.

Received January 22, 2019; Revised February 28, 2019; Accepted March 15, 2019

This paper is focused on the use of Approximated Computing (AC) for power consumption reduction. AC is a wide spectrum of techniques that relax the accuracy of computation in order to improve speed, energy, and/or another metric of interest. AC exploits the fact that several important applications do not necessarily need to produce precise results to be useful. Research interest in approximate computing has been growing in recent years, motivated by its potential in reducing power consumption. In this paper, we analyze the possibility to reduce power consumption in embedded digital Neural Networks (NN) using approximated algebraic operators in artificial neurons.

2. Low power Artificial Neuron Model

Figure 1 shows the block diagram of an artificial neuron. Such neuron is characterized by (2):

$$y = f(\sum x_i w_i + b) \quad (2)$$

where x_i and y represent respectively the inputs of the neuron and the output, w_i the weights, b the bias and f is the activation function (typically the sigmoid function).

Hardware implementation of digital neurons requires three main blocks:

- Multipliers: Used for the multiplication among the inputs and the weights
- Adders: Used to sum weighted inputs and bias
- ROM: Used for the implementation of the non-linear activation function

In several applications, the accuracy of the NN does not depend on the activation function topology. In such cases, the sigmoid function can be replaced by simpler functions as the satlins. A complexity reduction derives from this simplification. In these cases, ROMs can be replaced by simple multiplexers and comparators as shown in Figure 2. In the light of all these simplifications, the most complex block in terms of area and power consumption are the multipliers and the adders. For this reason, this paper is focused on reducing of the power consumption by replacing multipliers and adders with approximated operators [27].

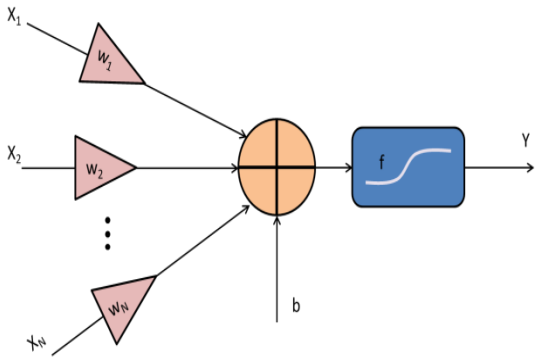

Figure 1. Artificial neuron model

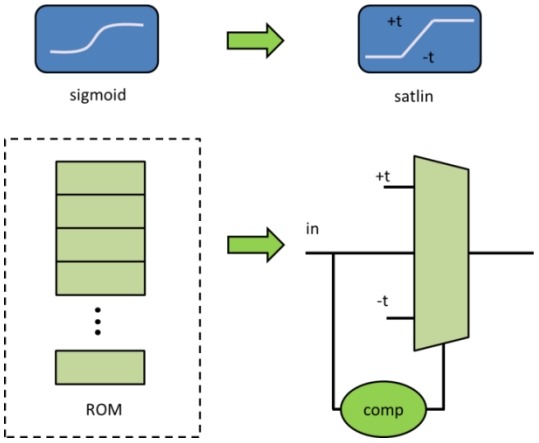

Figure 2. Linear approximation of a sigmoid activation function

2.1. Approximated Operators

As introduced in previous sections, in all the cases when sigmoid activation function can be replaced by a simple satlins, the power consumption of the artificial neuron depends essentially on the multiplier and the adders. In the following, we provide a description of the multipliers and adders used in our experiments. The Literature offers several approximated multipliers architectures. For our experiments, we use a modified version of the Compression based multipliers proposed in [28]. This multiplier works in three steps: partial products generation, partial products reduction and finally the sum of the reduced partial products using a

Carry Masked Adder (CMA), shown in Figure 3. More details about the architecture and the design of this multiplier are provided in [28]. In order to further reduce the area and consequently the power consumption, we replace the CMA with a sloppy adder [29]. This adder has been also used to realize the adder tree for the sum of the weighted inputs and the bias, as shown in Figure 4.

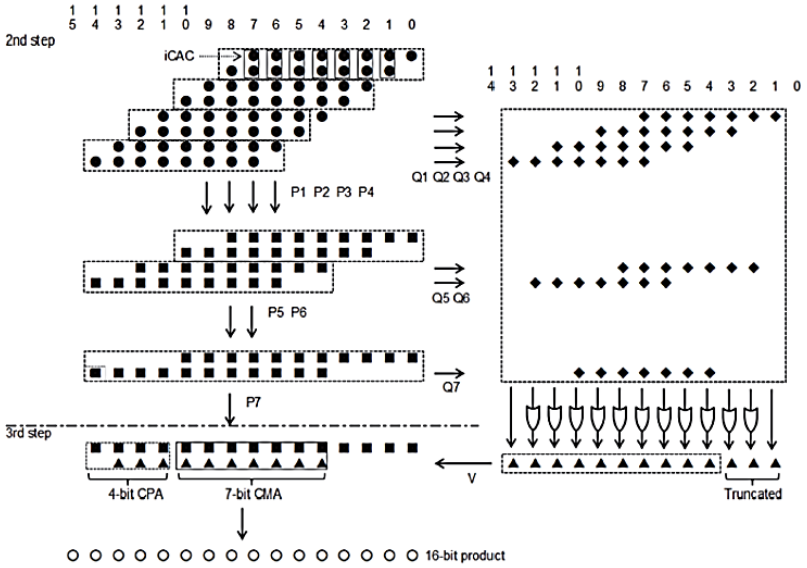

Figure 3. Carry masked addred (CMA) architecture

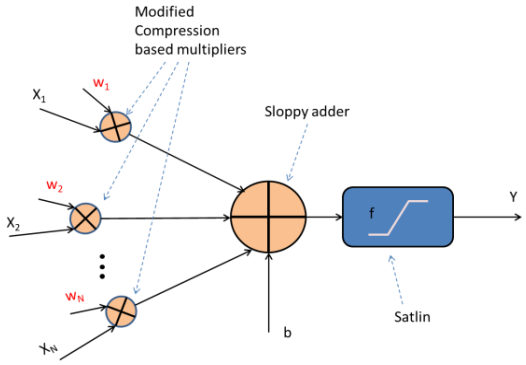

Figure 4. Approximated neuron models

3. Case of Study a Wine Classifier

In order to verify the performace of the proposed low power artificial neuron, it was tested in a NN. The case of study is the wine classifier available in the MATLAB Neural Networks tool box. The design and the training have been realized in MATLAB. The NN is composed of two layers: a hidden layer and an output layer shown in Figure 5. The hiddel layer is composed of 10 neurons having 13 input each while the output si composed by 3 neurons connected to the 10 outputs of the first layer. Inputs are wine features: Alcohol, Malic acid, Ash, Alkalinity of ash Magnesium, Total phenols, Flavanoids, Nonflavanoid phenols, Proanthocyanidins, Color intensity, Hue. The outputs are three different clesses of wine.

Experiments have been performed to estimate the power consumption reduction obtained by the approximated neuron model with respect to a traditional fixed-point neuron. Experiments followed this flow:

– We fixed the performances in terms of classification accuracy that the implemented hardware NN must achieve (we fixed the minimum accuracy to 95%)

- We designed the fixed-point architecture that assures such accuracy is respected.
- We designed the approximated architecture that respects such accuracy.
- The architectures obtained in point 3 and 4 are coded in VHDL and synthesized using Synopsys.
- The two architectures are characterized in terms of size area and speed.

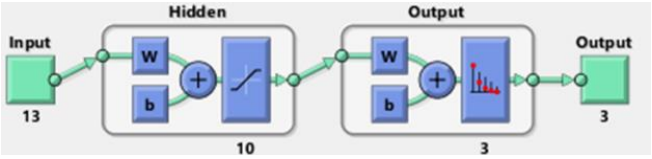

Figure 5. Multi-layer neural network for classification

To determine the number of bits required by the multipliers and the adders in the fixed-point architecture and in the approximated architecture, both the classifiers have been coded and simulated in MATLAB. The two models were developed using a tensorial representation, as in (3), in which multiplications and addition have been replaced with fixed-point operations and approximated operators respectively for the fixed-point and the approximated models.

$$\begin{bmatrix} i_1 \\ \vdots \\ i_{13} \end{bmatrix} * \begin{bmatrix} w_{1\,1} & \cdots & w_{1\,13} \\ \vdots & \ddots & \vdots \\ w_{1\,10} & \cdots & w_{10\,13} \end{bmatrix} \begin{bmatrix} b_1 \\ \vdots \\ b_{13} \end{bmatrix} \quad (3)$$

Simulations show that the fixed-point models meet the accuracy specification with the following parameters:

- 6-bit inputs 12-bit output multipliers
- 14-bit outputs adders

The approximated classifier meets the accuracy specification with the following parameters:

- 6-bit inputs, 11-bit output approximated multipliers
- 14-bit outputs adder with 5-bit sloppy

for both the models, inputs are 6 bits wide

4. Experimental Results

After the fixed-point analysis discussed in the previous section, we identified the most complex neuron (in terms of inputs number) in the classifier. Such neuron has been coded in VHDL in two different versions: the traditional fixed-point and the approximated one. For both the models, we use the satlin activation function. This is because, as discussed in previous sections, we focused our analysis on the algebraic operators. Both models are successively synthesized. The synthesis was performed using Synopsys Design Compiler and the STM 90 nm library of standard cells. Synthesis results are shown in Table 1 in terms of area, power consumption, and maximum frequency.

Results show that the approximated neuron outclasses the fixed-point model in area, power and maximum frequency. The area of the fixed-point neuron is 42280 μm^2 against the 23971 μm^2 of the approximated one. Such an advantage in area occupation involves less power consumption and higher clock frequency. The estimation of the area and the power consumption have been performed with a clock constraint equal to the maximum clock frequency obtainable by the fixed-point neuron (0.63 GHz).

Table 1. The Synthesis Results

	Approximate	Full	Saving
Area (at 0,63 GHz)	23971 μm^2	42280 μm^2	43 %
Power (at 0,63 GHz)	18,45 mW	28,57 mW	35 %
Max Frequency	0,76 GHz	0,63 GHz	20 %

5. Conclusions

In this paper, we investigated the possibility to realize low power Neural Networks using approximated algebraic operators. Experiments were performed comparing a traditional fixed-point neuron with an approximated neuron composed of a compression based multiplier and a sloppy adder. In both the neurons the satlin activation function has been used. Experiments show that in the proposed case of study (a wine classifier) the approximated neuron allows to save up to the 43% of the area, a power consumption saving of 35% and an improvement in the maximum clock frequency of 20%. Results underline that the use of approximated operators is a valid solution for power consumption and area occupation reduction in ML systems.

References

[1] Sciuto GL, Susi G, Cammarata G, Capizzi G. *A spiking neural network-based model for anaerobic digestion process*. 23rd International Symposium on Power Electronics, Electrical Drives, Automation and Motion (SPEEDAM). 2016: 996-1003.

[2] Brusca S, Capizzi G, Lo Sciuto G, Susi G. A new design methodology to predict wind farm energy production by means of a spiking neural network–based system*. International Journal of Numerical Modelling: Electronic Networks, Devices and Fields*. 2017; 7: e2267.

[3] Guadagni F, Zanzotto FM, Scarpato N, Rullo A, Riondino S, Ferroni P, Roselli M. RISK: A random optimization interactive system based on kernel learning for predicting breast cancer disease progression. Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics). 2017: 189-196.

[4] Ferroni P, Zanzotto FM, Scarpato N, Riondino S, Guadagni F, Roselli M. Validation of a machine learning approach for venous thromboembolism risk prediction in oncology. *Disease markers*. 2017.

[5] Fallucchi F, Zanzotto FM. Inductive probabilistic taxonomy learning using singular value decomposition. *Natural Language Engineering*. 2011; 17(1): 71-94.

[6] Fallucchi F, Zanzottto FM. *Singular value decomposition for feature selection in taxonomy learning*. Proceedings of the International Conference Recent Advances in Natural Language Processing (RANLP-2009). 2009: 82-87.

[7] Pazienza MT, Scarpato N, Stellato A, Turbati A. Semantic turkey: A browser-integrated environment for knowledge acquisition and management. *Semantic Web*. 2012; 3(3): 279-292.

[8] Ferroni P, Zanzotto FM, Scarpato N, Riondino S, Nanni U, Roselli M, Guadagni F. Risk assessment for venous thromboembolism in chemotherapy-treated ambulatory cancer patients: a machine learning approach. *Medical Decision Making*. 2017; 37(2): 234-242.

[9] Li S, Wen W, Wang Y, Han S, Chen Y, Li H. *An FPGA design framework for CNN sparsification and acceleration*. 2017 IEEE 25th Annual International Symposium on Field-Programmable Custom Computing Machines (FCCM). 2017: 28.

[10] Huang C, Ni S, Chen G. *A layer-based structured design of CNN on FPGA*. 2017 IEEE 12th International Conference on ASIC. 2017: 1037-1040.

[11] Cardarilli GC, Cristini A, Di Nunzio L, Re M, Salerno M, Susi G. *Spiking neural networks based on LIF with latency: Simulation and synchronization effects*. 2013 Asilomar Conference on Signals, Systems and Computers. Pacific Grove. 1838-1842.

[12] Khanal GM, Acciarito S, Cardarilli GC, Chakraborty A, Di Nunzio L, Fazzolari R, Cristini A, Re M, Susi G. Synaptic behaviour in ZnO-rGO composites thin film memristor. *Electronics Letters*. 2017; 53(5): 296-298.

[13] Khanal GM, Cardarilli G, Chakraborty A, Acciarito S, Mulla MY, Di Nunzio L, Fazzolari R, Re MA. *ZnO-rGO composite thin film discrete memristor*. (2016) IEEE International Conference on Semiconductor Electronics (ICSE). 2016: 129-132.

[14] Acciarito S, Cardarilli GC, Cristini A, Di Nunzio L, Fazzolari R, Khanal GM, Re M, Susi G. Hardware design of LIF with Latency neuron model with memristive STDP synapses. *Integration the VLSI Journal*. 2017; 59: 81-9.

[15] Acciarito S, Cristini A, Di Nunzio L, Khanal GM, Susi G. *An aVLSI driving circuit for memristor-based STDP*. 2016 12th Conference on PhD Research in Microelectronics and Electronics, PRIME. 2016: 1-4.

[16] Scarpato N, Pieroni A, Di Nunzio L, Fallucchi F. E-health-IoT universe: A review. *International Journal on Advanced Science, Engineering and Information Technology*. 2017; 7 (6): 2328-2336.

[17] Dalmasso I, Galletti I, Giuliano R, Mazzenga F. *WiMAX Networks for Emergency Management Based on UAVs*. IEEE–AESS European Conference on Satellite Telecommunications (IEEE ESTEL). 2012: 1-6.

[18] Giuliano R, Mazzenga F, Neri A, Vegni AM. Security access protocols in IoT capillary networks. *IEEE Internet of Things Journal*. 2017; 4(3): 645-657.

[19] Cardarilli GC, Di Nunzio L, Fazzolari R, Re M, Silvestri F, Spanò S. Energy consumption saving in embedded microprocessors using hardware accelerators. *TELKOMNIKA Telecommunication, Computing, Electronics and Control*. 2018; 16(3): 1019-1026.
[20] Cardarilli GC, Nunzio L, Fazzolari R, Giardino D, Matta M, Re M, Silvestri F, Spanò S. *Efficient Ensemble Machine Learning implementation on FPGA using Partial Reconfiguration.* International Conference on Applications in Electronics Pervading Industry, Environment and Society (IN PRESS). 2018.
[21] Giardino D, Matta M, Re M, Silvestri F, Spanò S*. IP Generator for Efficient Hardware Acceleration of Self-Organizing Maps*. International Conference on Applications in Electronics Pervading Industry, Environment and Society (IN PRESS). 2018.
[22] Cardarilli GC, Di Nunzio L, Fazzolari R, Giardino D, Matta M, Nannarelli A, Re M, Silvestri F, Spanò S. Comparison between Trigonometric, and traditional DDS, in 90 nm technology. *TELKOMNIKA Telecommunication Computing Electronics and Control*. 2018; 16(5): 2245-2253.
[23] Simonetta A, Paoletti MC. Designing Digital Circuits in Multi-Valued Logic. *International Journal on Advanced Science, Engineering and Information Technology*. 2018; 8(4): 1166-1172*.*
[24] Cardarilli GC, Di Nunzio L, Fazzolari R, Re M, Lee RB. *Integration of butterfly and inverse butterfly nets in embedded processors: effects on power saving*. 2012 Conference Record of the Forty Sixth Asilomar Conference on Signals, Systems and Computers (ASILOMAR). 2012: 1457-1459.
[25] Silvestri F, Acciarito S, Cardarilli GC, Khanal GM, Di Nunzio L, Fazzolari R, Re M. FPGA implementation of a low-power QRS extractor. *Lecture Notes in Electrical Engineering*. 2019; 512: 9-15.
[26] Weste N, Harris D. CMOS VLSI Design: A Circuits and System Perspective. 4th Edition. Addisin Wesley Publishing Company. 2010.
[27] M. Ammar Ben Khadra. An introduction to approximate computing. *arXiv*. 2017.
[28] Baba H, Yang T, Inoue M, Tajima K, Ukezono T, Sato T. *A Low-Power and Small-Area Multiplier for Accuracy-Scalable Approximate Computing*. 2018 IEEE Computer Society Annual Symposium on VLSI (ISVLSI). 2018: 569-574.
[29] Albicocco P, Cardarilli GC, Nannarelli A, Petricca M, Re M. *Imprecise arithmetic for low power image processing*. 2012 Conference Record of the Forty Sixth Asilomar Conference on Signals, Systems and Computers (ASILOMAR). 2012: 983-987.