EFFICIENT IMPLEMENTATION OF TANH: A COMPARATIVE STUDY OF NEW RESULTS

Samira Sorayaasa and Majid Ahmadi, FIEEE, FIET

Department of Electrical and Computer Engineering
University of Windsor, Windsor, Ontario, Canada

ABSTRACT

Hyperbolic tangent (Tanh) activation function is used in multilayered artificial neural networks (ANN). This activation function contains exponential and division terms in its expressions which makes its accurate digital implementation difficult. In this paper we present two different approximation techniques for digital implementation of Tanh function using power of two and coordinate rotation digital computer (CORDIC) methods. A comparative study of both techniques in terms of accuracy of their approximations in hardware costs as well as their speed when implemented on FPGA is also explained.

KEYWORDS

ANNs, Tanh, activation function, approximation with power of two, CORDIC algorithm, FPGA, optimization, hardware resource, latency, error.

1. INTRODUCTION

The widespread application of artificial neural networks has motivated researchers to implement ANNs on field programmable gate arrays (FPGA) for rapid prototyping. Although weight multiplication and signal summation in ANN are important, the most important part on a neural network is the activation function. Among all different types of activation functions, hyperbolic tangent is chosen (Tanh) because its ideal steep derivative, which allows a wider range of values for fast learning and its compatibility with derivative-based learning algorithm.

Tanh function is defined in equation (1) as:

$$Tanh(x) = \frac{e^{x} - e^{-x}}{e^{x} + e^{-x}} \qquad (1)$$

Physical implementation of equation (1) has two difficulties, exponential terms in the equation and division. There are many approaches to deal with problem associated with exponential approximation.

There is piecewise linear approximation [1], [2], [4].
Look up tables (LUTs) and range addressable look up tables (RALUTs) [3],[5].
Exponential approximation with power of two [6],[7].
Coordinate rotation digital computer (CORDIC) approximation [8],[9].
This paper presents a comparative study of the power of two and CORDIC approximation in terms of speed and resource utilization for their implementation and accuracy of the approximation.

David C. Wyld et al. (Eds): ARIN, CSITA, ISPR, NBIOT, DMAP, MLCL, CONEDU -2023
pp. 01-07, 2023. CS & IT - CSCP 2023 DOI: 10.5121/csit.2023.130701

To measure the accuracy of the approximation method RMSE of the function for original and approximated function has been calculated as

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(f(i)_a - f(i)_o)^2} \qquad (2)$$

Where n is the total number of samples and $f(i)_a$ and $f(i)_o$ are the approximated and the original output respectively.

Organization of this paper is as follows:

Section 2 presents the implementation of Tanh using power of two approximation, section 3 presents the approximation of Tanh using CORDIC method, section 4 presents the concluding and remarks.

2. EXPONENTIAL APPROXIMATION WITH POWER OF TWO

Tanh(x) shown in equation (1) can be simplified as

$$\text{Tanh}(x) = 1 - \frac{2e^{-x}}{e^{x} + e^{-x}} \qquad (3)$$

To reduce cost of e^x implementation conversion to power of two was recommended in [1] as shown in equation (4)

$$\exp(x) = 2^{1.44x} \qquad (4)$$

Two approximations of equation (4) are outlined in equations (5)&(6).

$$e^x \cong 2^{1.5x} = 2^{(1+\frac{1}{2})x} \qquad (5)$$

and

$$e^x \cong 2^{1.4345x} = 2^{\left(1+\frac{1}{2}-\frac{1}{8}+\frac{1}{16}\right)x} \qquad (6)$$

Table (1) shows the error of approximated Tanh with power of two as a function of the word lengths.

Table.1. Errors of approximated Tanh with power of two

Tanh	$e^x \cong 2^{(1+\frac{1}{2})x} = 2^{(1.5x)}$	$e^x \cong 2^{(2+1-\frac{1}{4}+\frac{1}{8})x}$
	RMSE	RMSE
16 bits	0.01242	0.0089
14 bits	0.01244	0.0089
8 bits	0.01432	0.0115

Implementation of the approximated Tanh with power of two has been carried out on Xilinx Vivado 2019 and simulated with Modelsim -Mentor edition and the simulation waveform editor included Modelsim altrn.

Table 2 shows the error of resource utilization of power of two approximation while table 3 shows the delay and power consumption of power of the two approximations.

Table .2. Hardware resource for power of two for 14 bits word lengths.

Tanh	$e^x \cong 2^{1.5x}$	$e^x \cong 2^{1.4375x}$
Multipliers	0	0
Add/Sub	3	4
Registers	0	0
Total 1 Bit Registers	0	0
RAMs	0	0
Mux	6	6
I/O Bits	28	28
Shifters	1	1

Table .3. Comparison of errors and resource utilization in power of two approximation

Approximation methods	LUT %	LUT RAM %	FF %	IO %	BUFG %	RMSE
Power of two $e^x \cong 2^{1.5x}$	0.16	0.01	0.02	2.75	3.13	0.0124
Power of two $e^x \cong 2^{1.4375x}$	0.16	0.01	0.02	2.75	3.13	0.0089

Table .4. Comparison of delay and power consumption on power of two approximation

Approximation methods	WNS (ns)	Delay (ns)	Time period (ns)	Frequency (MHz)	Max Frequency (MHz)	Total Power consumption(W)
Power of two $e^x \cong 2^{1.5x}$	0.225	9.775	10	100	102.30	0.140
Power of two $e^x \cong 2^{1.4375x}$	0.165	9.835	10	100	101.67	0.140

3. CORDIC ALGORITHM APPROXIMATION

Coordinate rotation digital approximation (CORDIC) [8],[10] is a low- cost iterative approximation and hardware friendly for evaluating various functions such as logarithm, exponential, trigonometric and hyperbolic functions.

The unified CORDIC algorithm iterative equations are shown below:

$$X_{i+1} = X_i - \beta\delta_i 2^{-i} y_i \qquad (7)$$

The β parameter shows that hardware is performing trigonometric, hyperbolic, and linear functions while the β value is chosen to be 1, -1 or 0 respectively.

$$Y_{i+1} = Y_i - \delta_i 2^{-i} X_i \qquad (8)$$

$$Z_{i+1} = Z_i - \delta_i F(i) \qquad (9)$$

Where $F(i)$ Represents the rotation function. Table .5 represents CORDIC parameter for various coordinate systems which represented in Fig.2.

Table .5.CORDIC operational modes and rotation functions [8]

β	Rotation type	$F(i)$
1	Circular Rotations (Sin, Cos, etc.)	$Tanh^{-1}(2^{-i})$
0	Linear Rotations [VLC] (Multiplication, Division)	(2^{-i})
-1	Hyperbolic Rotations [RHC] (Sinh,Cosh,etc)	$Tanh^{-1}(2^{-i})$

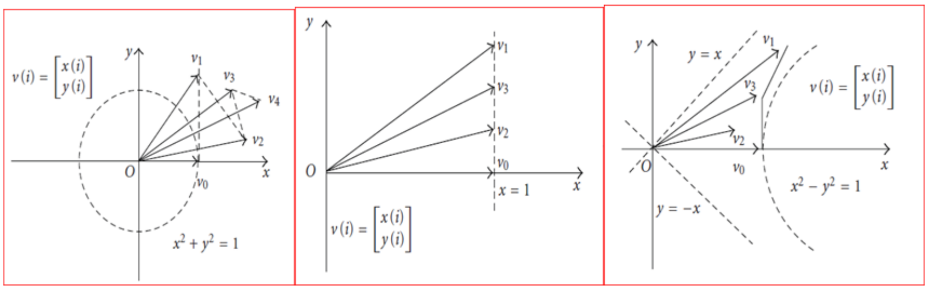

Fig.1. a. Circular system b. linear system c. hyperbolic system [10]

In this section we have implemented CORDIC algorithm to approximate Tanh function using 16,14,8 bits word lengths for any Angele. The proposed design is implemented using Xilinx vivado 2019 integrated development environment, and RTL code simulation test is carried out on Modelsim.

Simulation results and resource utilization and latency results verify the validity of the CORDIC algorithm in this design.

Table. 6 shows as number of iterations increases the error of approximation decreases.

Table.6. Absolute error for different number of iterations 1,3,5,7,9

NITERS	ERROR
1	0.55689331
3	0.19310669
5	0.0560669
7	0.01001831
9	0.00170044

Fig. 2 shows the pseudo code and HDL code for CORDIC approximation.

```
pseudo code for tangent Hyperbolic CORDIC.

1. function (CORDIC algorithm) determination.

      a. function Approximation. Problem=Tangent Hyperbolic

2. initialization of fixed-point arithmetic Data sample.

      b. Bound= (-4,4)

      c. word-length= [8,14,16]

      d. number of iterations= [5,15,25]

3. problem solving.

4. problem comparing

5. problem approximation.
```

Fig. 2. Pseudo code for CORDIC algorithm

Table .7. CORDIC approximation error in different word lengths

CORDIC Approximation	RMSE
16 Word lengths	0.00048995
14 Word lengths	0.0014719
8 Word lengths	0.035915

Table 8 shows the resource utilization of CORDIC approximation methods with14 bits word lenghts .

Table .8. resource utilization with 14 bits wordlenghts using CORDIC method

CORDIC method with 14 bits	
Multipliers	1
Add/Sub	107
Registers	89
Total 1 Bit Registers	1256
RAMs	0
Mux	49
I/O Bits	32
Shifters	12

4. CONCLUSION

In this paper we present two approximation methods for e^x along with their hardware implementations to be used for Tanh activation function. Our study shows power of two approximation is the most hardware efficient technique. While CORDIC methods gives the best delay results. For the accuracy of the two approximation methods CORDIC algorithm shows superiority to power of two approximation methods.

REFERENCES

[1] Gomperts, A., Ukil, A., & Zurfluh, F. "Development and implementation of parameterized FPGA-based general purpose neural networks for online applications" IEEE Transactions on Industrial Informatics, 7(1), pp.78-89, 2010.

[2] Basterretxea, K., Tarela, J. M., & Del Campo, I. "Approximation of sigmoid function and the derivative for hardware implementation of artificial neurons" IEE Proceedings-Circuits, Devices and Systems, 151(1), pp.18-24, 2004.

[3] Leboeuf, K., Namin, A. H., Muscedere, R., Wu, H., & Ahmadi, M. "High speed VLSI implementation of the hyperbolic tangent sigmoid function" 2008 Third international conference on convergence and hybrid information technology, Vol. 1, pp. 1070-1073.

[4] Wei, L., Cai, J., Nguyen, V., Chu, J., & Wen, K. "P-SFA: Probability based sigmoid function approximation for low-complexity hardware implementation" Microprocessors and Microsystems, 76, pp.103-105, 2020.

[5] Muscedere, R., Dimitrov, V., Jullien, G. A., & Miller, W. C. "Efficient techniques for binary-to-multidigit multidimensional logarithmic number system conversion using range-addressable look-up tables" IEEE Transactions on Computers, 54(3), pp. 257-271,2005.

[6] Gomar, S., Mirhassani, M., & Ahmadi, M. "Precise digital implementations of hyperbolic tanh and sigmoid function" 50th Asilomar Conf. on Signals, Systems and Computers pp. 1586-1589, 2016.

[7] Gomar, S., & Ahmadi, A. "Digital multiplier less implementation of biological adaptive-exponential neuron model" IEEE Transactions on Circuits and Systems I: Regular Papers, 61(4), pp.1206-1219, 2013.

[8] Chen, H., Jiang, L., Luo, Y., Lu, Z., Fu, Y., Li, L., & Yu, Z. "A CORDIC-Based Architecture with Adjustable Precision and Flexible Scalability to Implement Sigmoid and Tanh Functions" 2020 IEEE- ISCAS, pp. 1-5, 2020.

[9] Nguyen, T. D., Kim, D. H., Yang, J. S., & Park, S. Y. "High-Speed ASIC Implementation of Tanh Activation Function Based on the CORDIC Algorithm" 2021, 36th Int Tech Conf. on Circuits/Systems, Computers and Communications (ITC-CSCC) pp. 1-3,2021.

[10] Chen, Y. H., Chen, S. W., & Wei, M. X. (2020) "A VLSI implementation of independent component analysis for biomedical signal separation using CORDIC engine" IEEE transactions on biomedical circuits and systems, 14(2), pp 373-381.

AUTHORS

Samira Sorayaasa Received BSc from Jahron University in 2008, MSc in Management from Shiraz University in 2012, MSc from University of Windsor 2022. She is IEEE member.

Majid Ahmadi, Received BSc from Sharif University in 1970, PhD from Imperial College in 1977. He has been with the Department of ECE, University of Windsor since 1980. He holds the rank of Distinguished Professor. He is FIET and FIEEE.