2012 IEEE fifth International Conference on Advanced Computational Intelligence(ICACI)
October 18-20, 2012 Nanjing, Jiangsu, China

A Non-linear Approximation of the Sigmoid Function based on FPGA

Zhenzhen Xie

***Abstract*—One of the difficult problems encountered when implementing artificial neural networks based on FPGA is the approximation of the activation function. The sigmoid function is the most widely used and is difficult to approximate. This paper is devoted to show a saving hardware resources and accurate way to compute the sigmoid function based on FPGA by non-linear approximation. This is done by subsection analysis involved a new low-leakage FPGA Look-up Tables (LUTs), introducing a non-linear approximation algorithm in detail, analyzing the approximating accuracy and the FPGA hardware resources, which can achieve some kind of balance between the approximating precision and the limited hardware resources of FPGA, shows improvements over the previous known algorithms. The implementation of sigmoid function and the simulation are completed by the development software of QUARTUS II.**

I. Introduction

SPECIFIC hardware is considered to be a better solution than the most common implementation of Artificial Neural Networks (ANN) with a personal computer (PC). During the varieties of specific hardware solutions the FPGA solution is the most interesting implementation of Artificial Neural Networks, considering the need for higher processing speed, reduced cost for each implementation and reliability. The biggest difficulty of this approach lies in how to achieve high-precision ANN network using the limited FPGA hardware resources, while, the implementation of neuronal activation function using FPGA hardware has always been a great challenge.

There are many kinds of neuron activation functions, of which the sigmoid function is the most widely used.

As authors in the literatures [1]-[3] have studied on approximation algorithms of sigmoid function and point that the solution of truncation of Taylor series expansion has the lowest approximation accuracy comparing other classical solutions including the look-up tables and CORDIC (Coordinated Rotation Digital Computer) those can achieve higher accuracy but occupy too much hardware resources of FPGA. In the literatures [4],[5] another algorithm is proposed--- piecewise-linear approach, which requires less hardware resources and has a higher processing speed, whereas has relatively larger error.

Referred the literatures [1]-[5] a Non-linear Approximation combined look-up Tables approach is presented, it has the benefit of the solutions in literatures[2-5]

Z Xie is with the department of electrical engineering, Zhicheng college of Fuzhou University, Fuzhou 350007,Fujian PRC(phone:15005914893;e-mail:xzzwin@163.com).

II. The Non-linear Approximation Algorithm

Log-sigmoid function, also known as a logistic function, is given by the graph of a sigmoid function in Fig.1, it will be easily found has three features: First, it models the S-shape curve; Second, when the independent variable range over the real numbers from -∞ to +∞ it tends to 0 and 1; Third, it is linear trend on the open interval(-1,1). Make use of the Polynomial curve fitting function in MATLAB – polyfit function, take the independent variable span of 0.0001, and separate the sigmoid function into twenty-five piece-wise functions to approximate, as is shown in Table 1.

TABLE I
THE INTERVAL/FUNCTION FORM

Interval	Function Form
x (-1,1)	y=0.2383x+0.5000
x [-2,-1]	y=0.0467* x^2 +0.1239*x+0.2969
x [-3,-2)	y=0.0298* x^2 +0.2202*x+0.4400
x [-4,-3)	y=0.0135* x^2 +0.1239*x+0.2969
x [-5,-4)	y=0.0054* x^2 +0.0597*x+0.1703
x [-5.03,-5)	y=0.0066
x [-5.2,-5.03)	y=0.0060
x [-5.41,-5.2)	y=0.0050
x [-5.66,-5.41)	y=0.0040
x [-6,-5.66)	y=0.0030
x [-6.53, -6)	y=0.0020
x [-7.6,-6.53)	y=0.0010
x [-∞,-7.6)	y→0
x [1,2)	y=-0.0467* x^2 +0.2896*x+0.4882
x [2,3)	y=-0.0298* x^2 +0.2202*x+0.5600
x [3,4)	y=-0.0135* x^2 +0.1239*x+0.7030
x [4,5)	y=-0.0054* x^2 +0.0597*x+0.8297
x [5,5.0218)	y=0.9930
x [5.0218,5.1890)	y=0.9940
x [5.1890,5.3890)	y=0.9950
x [5.3890,5.6380)	y=0.9960
x [5.6380,5.9700)	y=0.9970
x [5.9700,6.4700)	y=0.9980
x [6.4700,7.5500)	y=0.9990
x [7.5500,+∞)	y→1

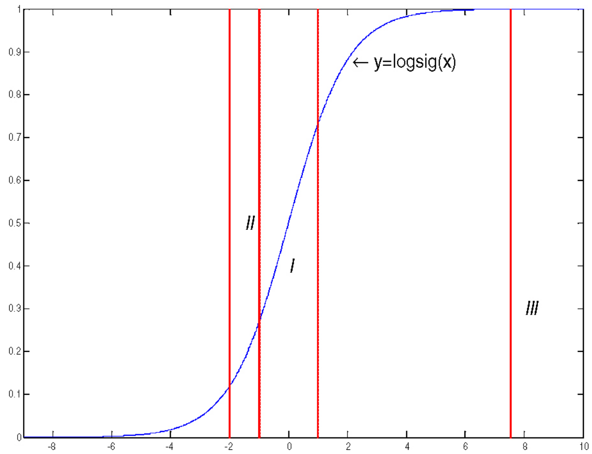

Fig. 1. Sigmoid function

Take three representative segments of sigmoid approximation to analyze. In the segment I ,as marked in the Fig.1,the sigmoid function is near the linearity in the interval(1,1), so using the linear function as in Equation (1) to approximate it.

$$y=0.2383x+0.5000 \quad (1)$$

The absolute error of this segment is 0.0073, and the sigmoid function curve of this part can be seen in the Fig.2, where the approximation curve has been drawn in dotted line.

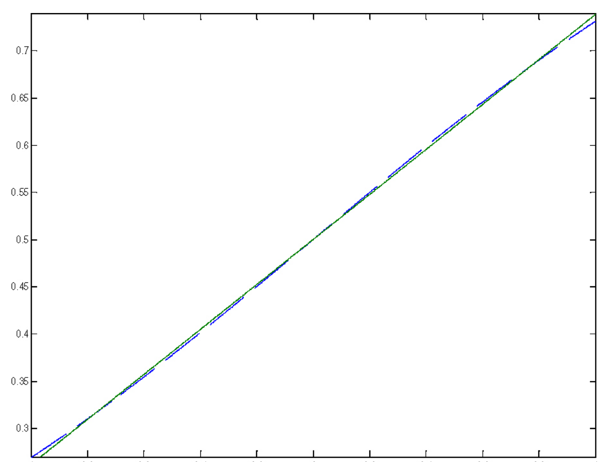

Fig. 2. The Approximation of Segment I

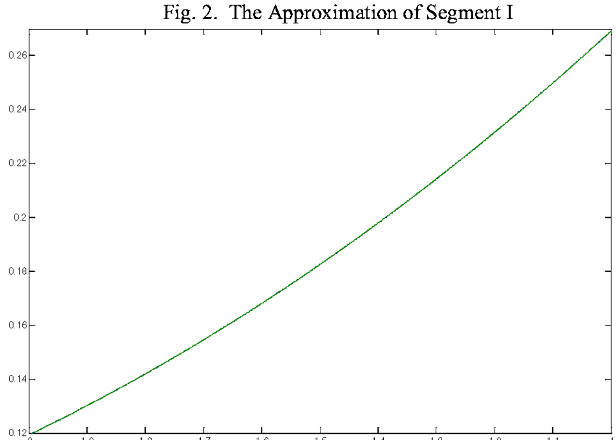

Fig. 3. The Approximation of Segment II

In the segment II , as marked in the Fig.1, through testing by polyfit function we find during the interval [-2,-1] ,the sigmoid function can be finely approximated in the non-linear function as in Equation(2),

$$y=0.0467*x^2+0.1239*x+0.2969 \quad (2)$$

The absolute error of this segment is 4.8230e-005, this part of sigmoid function curve can be seen in the Fig.3, where the approximation curve has been drawn in dotted line.

As is known that the larger degree of polynomial the higher accuracy of the approximation gained, that is the approximation error will be smaller. In view of the limited hardware resources in FPGA and the demand of accuracy of ANN based FPGA the quadric polynomial is adopted.

In the segment III, its Positive infinity tends to 1 and its negative infinity tends to 0, so chose the linear function as in Equation (3) to approximate the sigmoid function. Besides, the terminal point can be expanded if necessary. In this segment, 7.55 is chose as the left terminal point.

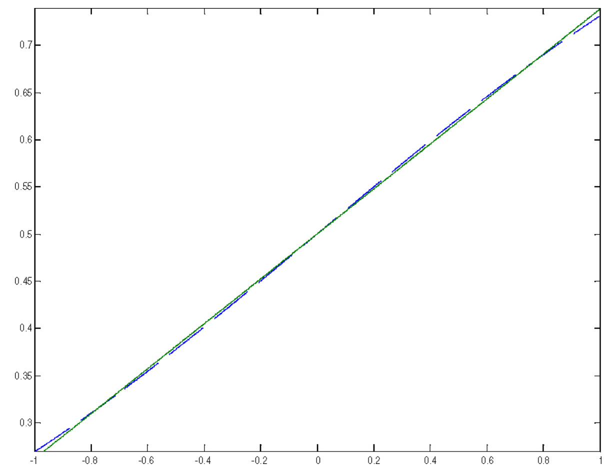

Fig. 4. The Approximation of Segment II

$$y=1 \quad (3)$$

The absolute error of this segment is 0.0005, and the sigmoid function curve of this part can be seen in the Fig.4, where the approximation curve has been drawn in dotted line.

When the value of sigmoid function has a range of 0.0005, such as the interval [-5.03,-5), [-5.2,-5.03) and so on, all can be found in the TABLE I, the value of the certain interval will be stored into RAM and later be read directly if necessary. The absolute errors of these intervals are no more then 0.001.

The absolute error with each segment of sigmoid function as is shown in TABLE II, which is computed by the tool of MATLAB.

TABLE II
INTERVALS/ABSOLUTE ERROR

Interval	Absolute Approximate Error Table
x (-1,1)	0.0073
x [-2,-1]	4.8230e-005
x [-3,-2)	6.1374e-006
x [-4,-3)	1.0617e-005
x [-5,-4)	5.7802e-006
x [-5.03,-5)	0.0002
x [-5.2,-5.03)	0.0010
x [-5.41,-5.2)	0.0010
x [-5.66,-5.41)	0.0010
x [-6,-5.66)	0.0010
x [-6.53, -6)	0.0010
x [-7.6,-6.53)	0.0010
x [-∞,-7.6)	5.0020e-004
x [1,2)	4.8213e-005
x [2,3)	6.1326e-006
x [3,4)	1.0609e-005
x [4,5)	9.1024e-005
x [5,5.0218)	0.0002
x [5.0218,5.1890)	0.0010
x [5.1890,5.3890)	0.0010
x [5.3890,5.6380)	0.0010
x [5.6380,5.9700)	0.0010
x [5.9700,6.4700)	0.0010
x [6.4700,7.5500)	0.0010
x [7.5500,+∞)	0.0005

III. THE IMPLEMENTATION OF NON-LINEAR APPROXIMATION ALGORITHM

The implementation of non-linear approximation algorithm and the simulation are completed by the development software of QUARTUS II, using the VHDL. The test data are represented by 16 bit fixed-point decimal. The simulation result as is shown in Fig.5, the simulation result, the theoretical value and the absolute error are listed in TABLE III.

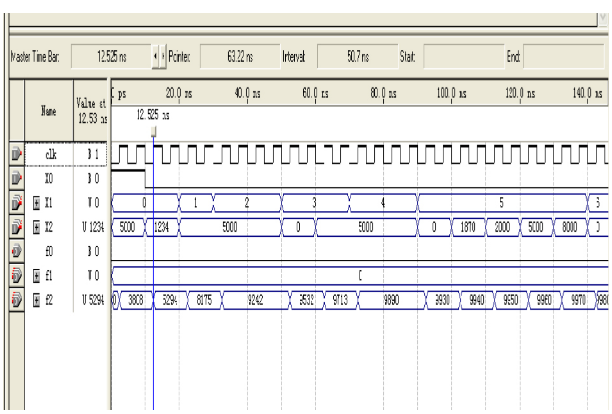

Fig. 5. The Simulation Result of the Implementation of Sigmoid Function

The result of the implementation of sigmoid function shows that the absolute errors between the simulation value based FPGA and theoretical value based MATLAB are no more than 0.0033, which shows the non-linear approximation of sigmoid function is successfully accomplished in FPGA implementation.

TABLE III
THE ANALYSIS OF THE SIMULATION RESULT

Argument of Sigmoid Function in the Simulation	Simulation Value of the Sigmoid Function based QuartusII	Theoretical Value of the Sigmoid Function based MATLAB	Absolute Error between Column2 and Column3
-0.5000	0.3808	0.3775	0.0033
0.1234	0.5308	0.5294	0.0014
1.5000	0.8175	0.8176	0.0001
2.5000	0.9242	0.9241	0.0001
3.0000	0.9532	0.9526	0.0006
3.5000	0.9713	0.9707	0.0006
4.5000	0.9890	0.9890	0
5.000	0.9930	0.9933	0.0003
5.1870	0.9940	0.9944	0.0004
5.2000	0.9950	0.9945	0.0005
5.5000	0.9960	0.9959	0.0001
5.8000	0.9970	0.9970	0
6.0000	0.9980	0.9975	0.0005

IV. CONCLUSION

Piece-wise non-linearization can be used for finely approximating sigmoid function as increasing the accuracy with relatively less hardware overhead. Furthermore, if the hardware resources of FPGA are rich the approximation could be improved in two ways: one is to divide the sigmoid function into more segments; the other is using the higher degree of polynomial to approximate.

REFERENCES

[1] Z. Lu, “Analysis and Comparison of Approximation Capability in Several Methods”, *Journal of Wuzhou University*, 3, 18, pp.13-17, Sep.2008.

[2] H. Zhang, X. Li and S. Tian, “Simulation Line Design and Its FPGA Realization Based on BP Neural Network”, *Journal of Electronics & Information Technology*, 5, 29, pp.1267-1270, 2007.

[3] J. Valls, “Evaluation of CORDIC Algorithms for FPGA Design”, Journal of VLSI Signal Processing, 32, pp. 207-222. 2002.

[4] P. Ferreira, P. Ribeiro and A. Antunes, F. M. Dias, “Artificial Neural Networks Processor- A Hardware Implementation Using a FPGA”. In International Conference on Field Programmable Logic and Applications, 2004, pp. 0830-0901.

[5] X. Xie, B. Yang, Z. Chen, “a FPGA based Artificial Neural Network Prediction Approach”, Shang Dong Science, 1, 22, pp. 7-12,2009, 22(1):7-12.