New Optimization Models for Sine Cosine Functions in Embedded Telecommunication Systems

Mohamed Saber[1*], El-Sayed M. El-Kenawy[2], Abdelhameed Ibrahim[3], Marwa M. Eid[4], Abdelaziz A. Abdelhamid[5]

[1]Electronics and Communications Engineering Dep., Faculty of Engineering, Delta University for Science and Technology, Gamasa City, Mansoura, Egypt

[2]Department of Communications and Electronics, Delta Higher Institute of Engineering and Technology, Mansoura, 35111, Egypt

[3]Computer Engineering and Control Systems Department, Faculty of Engineering, Mansoura University, 35516, Mansoura Egypt

[4]Faculty of Artifcial Intelligence, Delta University for Science and Technology, Mansoura, Egypt

[5]Computer Science Department, Faculty of Computer and Information Sciences, Ain Shams University, Cairo, 11566, Egypt

Emails: Mohamed.saber@deltauniv.edu.eg; skenawy@ieee.org; afai79@mans.edu.eg; mmm@ieee.org; abdelaziz@cis.asu.edu.eg

Abstract

Trigonometric functions are essential part of digital communication systems such as receivers, synthesizers, and phase locked loop. Implementation of trigonometric functions requires many arithmetic units; multipliers and adders circuits which reduces the speed of operation and consumes much power. In this paper we introduce two approximation methods to represents sine, and cosine functions to achieve fast operation and low power consumption. The simulations indicate a matching between the ideal trigonometric functions and the approximation method with 0.001 error which considered as trivial amount.

Keywords: trigonometric functions approximation; low power consumption; FPGA.

1. Introduction

The approximation of trigonometric functions is an important topic in electronics engineering, including digital communications, arithmetic computing, and signal processing [1–4]. Field programmable gate arrays (FPGAs) and micro controllers are used to implement digital circuits suffers from limited resources such as floating-point multipliers, dividers, adders, LUT, and digital signal processors. Therefore, simplification techniques are required to reduce the large number of calculations in embedded systems [5] for example, two arithmetic operations; division and square root are slower than multiplication and addition [6,7].

Small, low-power programmable logic circuits are utilized in many goods and applications, including smartphones, calculators, audio accessories, medical equipment, Internet of Things devices, and remote-control drones [8-19]. Fast approximate computation is also crucial in machine learning for addressing issues like accelerated learning and immediate inference processing [20-23].

Improving performance and reducing model size in machine learning is feasible by performing approximation calculations using multiply-add operations, which can be processed in parallel by graphics processing units.

Doi : https://doi.org/10.54216/IJWAC.030205
Received: July 01, 2021 Accepted: August 30, 2021

When the performance of processor is low, reducing the operation are required to be suitable with the speed of the processor such as replacing the division process with another mathematical operations [24-25]. Also, different techniques are used to implement exponential and logarithmic functions [21-26], and reciprocal sqrt.

Taylor expansions is the first approximations of trigonometric functions; however, it suffers from slow conversions or long computation time. While inverse trigonometric functions replaced by many methods such as LUT, CORDIC algorithm, rational, polynomial approximations. Unfortunately, all the mentioned methods require large amount of memory.

This paper presents a proposed method to approximate sine, cos, and inverse tan functions depending on adder and multiplier without the use of a memory. The optimization method minimizes the error over the entire domain.

2. Proposed Mathematical Method

A- First Method

To do rapid calculations for embedded systems, the suggested method implements the idea of encoding sine and cosine functions using a mathematical formula based on multiplication and addition without memory, division operations, and iterative convergence procedures. The approximate formula was developed using the steps below.

1- Linear approximation with fixed origin and domain endpoints:

$$g(t) \approx \alpha t + \beta$$

2- Evaluate residuals with approximate target function:

$$e_{res} = \alpha t + \beta - g(t)$$

3- Design of residual correction function (RCF):

$$r(t) = e_{res}$$

4- Approximate formula with residual correction:

$$f_{approx}(t) = \alpha t + \beta - r(t)$$

Figure 1: RCF design procedure

For sine and cosine functions, it was demonstrated that an effective approximation formula might be obtained using parabolic functions, considering both processing cost and approximation error. Some tasks, however, necessitate a more precise result. The second approach involves extending the approximation formula with the residual correction to lower the approximated error while keeping the processing cost constant. Residual correction approximations for trigonometric functions are as follows:

$$\sin(\theta) \approx s(\theta) = \left(\frac{2}{\pi}\right)^2 \theta(\pi - |\theta|) \qquad (1)$$

$$\cos(\theta) \approx c(\theta) = s(\frac{\pi}{2} - |\theta|) \qquad (2)$$

These are of the same form as the famous quadratic approximation for a sine function in $[-\pi{:}\,\pi]$, but the error is larger around $\theta = \frac{\pi}{4}$, as shown in Figure 2.

Doi : https://doi.org/10.54216/IJWAC.030205
Received: July 01, 2021 Accepted: August 30, 2021

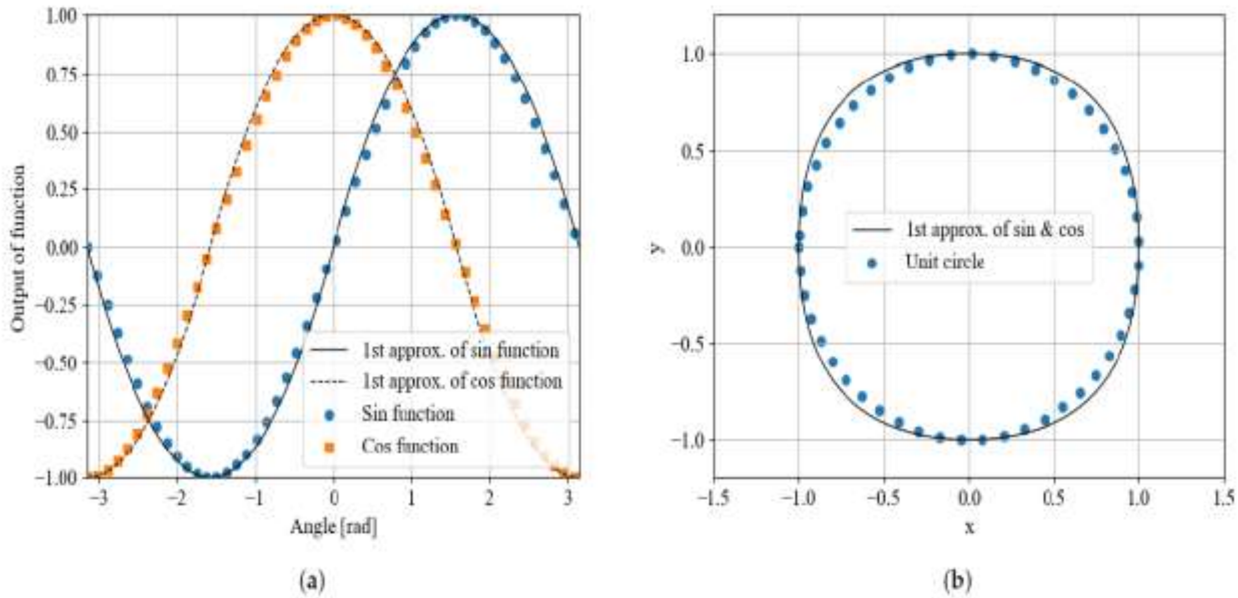

Figure 2: (a) Approximation of sin and cos functions by RCM. (**b**) Comparison with unit circle.

To carry out the procedure while being limited to the multiply-add method of computation. For this purpose, we consider the sin function and assess the initial approximation's residuals. For this residual, we can consider the modified RCF that is obtained by multiplying and adding:

1- RCF with signed quadratic function:

$$r_{sq}(\theta) = \alpha_{sq} s(\theta)(1 - |s(\theta|)$$

2- RCF with cubic function:

$$r_{cubic}(\theta) = \alpha_{cubic}\, s(\theta)\left(1 - s^2(\theta)\right)$$

3- RCF with co-function:

$$r_{cofunc}(\theta) = \alpha_{cofunc}\left(s(\theta) - sign\left(s(\theta)\right)\sqrt{1 - c^2(\theta)}\right)$$

Figure 3: 1st method procedure

B. Second Method

Using $r_{sq}(\theta)$ for RCF, the second method of the sine function is

$$s_2(\theta) = s(\theta) - r_{sq}(\theta) = s(\theta)\left[\left(1 - \alpha_{sq}\right) - \alpha_{sq}|s(\theta)|\right] \quad (3)$$

A similar argument can be made for the second method of the cos function:

$$c_2(\theta) = c(\theta)\left[\left(1 - \alpha_{sq}\right) - \alpha_{sq}|c(\theta)|\right] \quad (4)$$

The approximate shape of the final second approximation function is shown in Figure 4.

Doi : https://doi.org/10.54216/IJWAC.030205
Received: July 01, 2021 Accepted: August 30, 2021

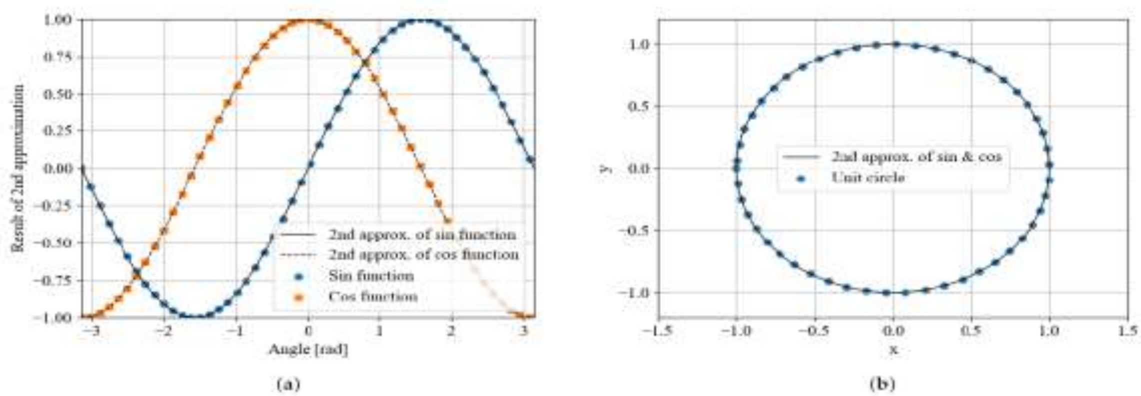

Figure 4: (**a**) 2nd method of sin and cos functions by RCM. (**b**) Comparison with unit circle.

3. Results

To investigate the performance of the 1st, 2nd approximation methods by showing the residual error, and execution time of both methods. Figure 5 shows the residual error of the first of 1st approximation method it approximately equals 0.056. figure 6. Shows the residual error of the 2nd method which is reduced to be 0.000092.

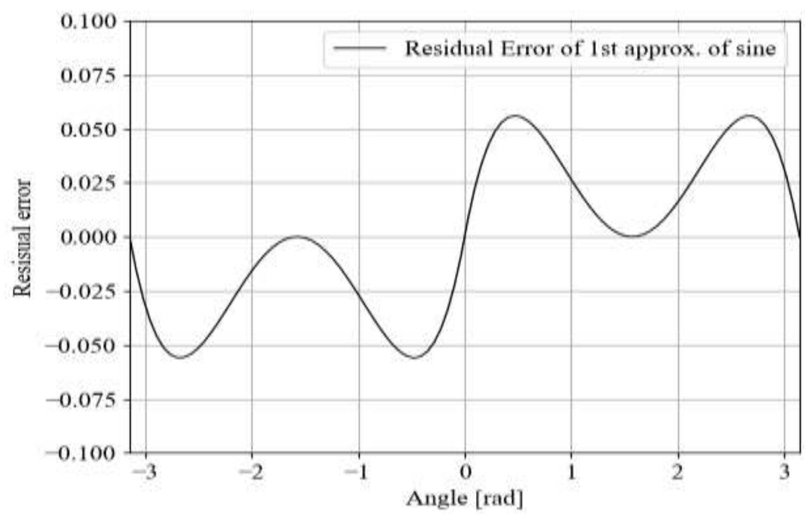

Figure 5: Residual errors of the 1st approximation method

Doi : https://doi.org/10.54216/IJWAC.030205
Received: July 01, 2021 Accepted: August 30, 2021

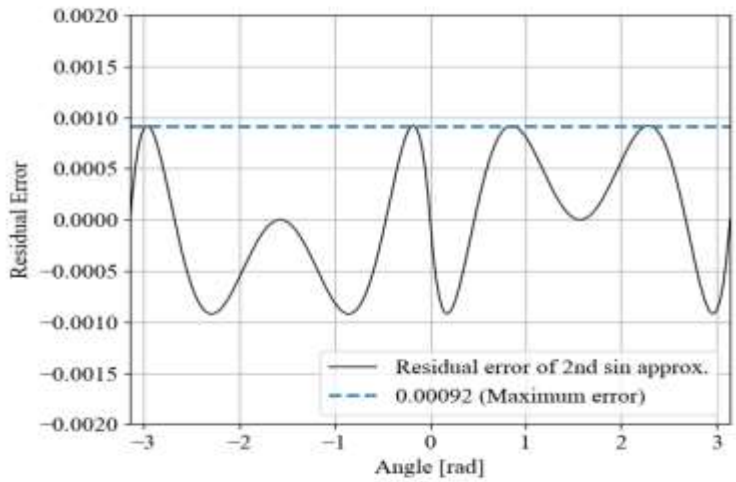

Figure 6: Residual error of 2nd approximation of sin-function.

Table 1 indicates the ecxuation time of both approximation methods and the

As seen in figure 6. The final residual is, at most, approximately 0.00092, which is sufficiently small when compared with the first approximation method.

Table 1: Execution time of 1st and 2nd approximation methods

Method		Execution time (ns)
Sine function	1st method	0.54 ± 0.16
	2nd method	1.4 ± 0.2
Cosine function	1st method	1.22 ± 0.16
	2nd method	2.04 ± 0.39
Math library (math.h)	Sine function	68.28 ± 0.52
	Cosine function	69.1 ± 0.9

The processing time was measured on a 2.5 GHz Intel Xeon E3 processor. The result was an approximation for sine that took 1.4 ns to calculate and an approximation for cosine that took 2.04 ns to figure; both were roughly twice as fast as the initial approximation but still over 30 times faster than the standard mathematical built-in functions. h. The computational cost and maximum error for the first and second approximation approaches are shown in Table 2.

Table 2: Error and computational cost of approximating trigonometric functions.

Sine Function	Maximum Error	Computational Cost
1st method	5.6×10^{-2}	1 addition 2 multiplications 1 absolute value
2nd method	9.2×10^{-4}	2 additions 4 multiplications 2 absolute values

Table 3 and Figure 7 show the comparison results between the Taylor series and both approximation methods.

Doi : https://doi.org/10.54216/IJWAC.030205
Received: July 01, 2021 Accepted: August 30, 2021

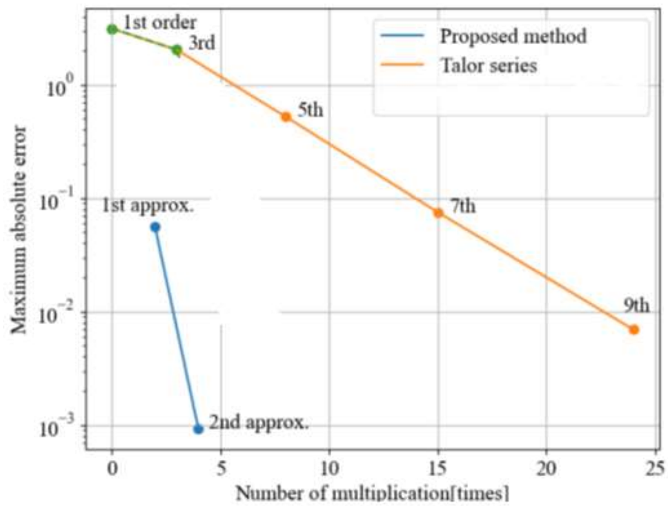

Figure 7: Comparison of maximum error and number of multiplications of sin function approximation between Taylor series and proposed two methods

As shown in figure 7 Taylor series required several iterations to achieve accurate results. The 9th iteration of Taylor series has error more than the 2nd approximation method.

Table 3: Comparison between Taylor series and two proposed methods

	Taylor series (iterations)					Proposed method	
	1st	3rd	5th	7th	9th	1st	2nd
Maximum absolute error	3.14	2.03	5.24×10^{-1}	7.52×10^{-2}	6.93×10^{-3}	5.6×10^{-2}	9.2×10^{-4}
Number of multiplication	0	3	8	15	24	2	4

As shown in Table 3. The 1st method have a smaller error compred to taylor series with 9th iterations while the 2nd method achives a less error compared to 1st method. Also the number of multiplications increases with the number of iterations in taylor series method but with the propsed 1st method achives two multiplication only and four in the 2nd proposed method.

4. Conclusions

In this paper, we proposed two approximation methods to sine, and cosine with a high accuracy using a small number of calculations to allow simple implementation for embedded systems. Simulations indicate a reduction of absolute error in the 2nd method to be 0.000092 which is negligible. Also, the proposed method reduces the number of multiplications to be 2 in the 1st method and 4 in the 2nd method which reduced the power consumption and increase4 the operating frequency. A comparison with Taylor series indicates that the proposed system achieves less error compared to the 9th iteration of Taylor series. The proposed method suitable for embedded systems in drones and robots.

References

[1] Moroz, L.; Samotyy, V. Efficient Floating-Point Division for Digital Signal Processing Application [Tips &Tricks]. IEEE Signal Process. Mag. 2019, 36, 159–163.

[2] M. Saber, M. Elmasry, M. E. Abo-Elsoud, “Quadrature Direct Digital Frequency Synthesizer Using FPGA”, International Conference on Computer Engineering and systems, pp.14-18, Nov.

Doi : https://doi.org/10.54216/IJWAC.030205
Received: July 01, 2021 Accepted: August 30, 2021

[3] M. Saber, Y. Jitsumatsu and M.T.A. Khan, “A Low Power Digital Phase Locked Loop with Rom-free Numerically Controlled Oscillator," Signal Processing: An international journal (SPIJ), vol.5, no.4, pp.142-155, Oct. 2011.

[4] Saber, M.; Jitsumatsu, Y.; Kohda, T. A low-power implementation of arctangent function for communication applications using FPGA. In Proceedings of the 2009 Fourth International Workshop on Signal Design and its Applications in Communications, Fukuoka, Japan, 19–23 October 2009; pp. 60–63.

[5] Oberman, S.; Flynn, M. Design issues in division and other floating-point operations. IEEE Trans. Comput. 1997, 46, 154–161.

[6] M. Saber, Y. Jitsumatsu and M.T.A. Khan, “Design and Implementation of Low Ripple Low Power Digital Phase-Locked Loop", Signal Processing: An international journal (SPIJ), vol.4, no.6, pp.304-317, Feb. 2011.

[7] M. Saber, Y. Jitsumatsu, Y. M.T.A. Khan, ” A Simple Design to Mitigate Problems of Conventional Digital Phase Locked Loop ”, Signal Processing: An international journal (SPIJ), vol.6, no.2, pp.65-77, April. 2012.

[8] Bolanakis, D.E. A Survey of Research in Microcontroller Education. IEEE Rev. Iberoam. Tecnol. Del Aprendiz. 2019, 14, 50–57.

[9] Buturug˘a, A.; Constantinescu, R.C.; Stoichescu, D.A. Current consumption analysis for 8-bit microcontrollers. In Proceedings of the 2019 11th International Conference on Electronics, Computers and Artificial Intelligence (ECAI), Pitesti, Romania, 27–29 June 2019; pp. 1–6.

[10] Chen, C.H.; Lin, M.Y.; Liu, C.C. Edge Computing Gateway of the Industrial Internet of Things Using Multiple Collaborative Microcontrollers. IEEE Netw. 2018, 32, 24–32.

[11] Joo, S.; An, Y.J.; Oh, T.W.; Jung, S.O. Comparative analysis of MCU memory for IoT application. In Proceedings of the 2018 International Conference on Electronics, Information, and Communication (ICEIC), Honolulu, HI, USA, 24–27 January 2018; pp. 1–3.

[12] Cococcioni, M.; Rossi, F.; Ruffaldi, E.; Saponara, S. Fast Approximations of Activation Functions in Deep Neural Networks when using Posit Arithmetic. Sensors 2020, 20, 1515.

[13] Bouguezzi, S.; Faiedh, H.; Souani, C. Hardware Implementation of Tanh Exponential Activation Function using FPGA. In Proceedings of the 2021 18th International Multi-Conference on Systems, Signals Devices (SSD), Monastir, Tunisia, 22–25 March 2021; pp. 1020–1025.

[14] Chang, C.H.; Chen, S.H.; Chen, B.W.; Wang, J.C.; Wang, J.F. A division-free algorithm for fixed-point power exponential function in embedded system. In Proceedings of the 2013 1st International Conference on Orange Technologies (ICOT), Tainan, Taiwan, 12–16 March 2013; pp. 223–226.

[15] Ibrahim, Abdelhameed, and El-Sayed M. El-kenawy. "Applications and datasets for superpixel techniques: A survey." Journal of Computer Science and Information Systems 15, no. 3 (2020): 1-6.

[16] Moroz, L.V.; Samotyy, V.V.; Horyachyy, O.Y. Modified Fast Inverse Square Root and Square Root Approximation Algorithms: The Method of Switching Magic Constants. Computation 2021, 9, 21.

[17] El-Sayed Towfek, M., and M. Saber El-kenawy. "Reham Arnous. An Integrated Framework to Ensure Information Security Over the Internet." International Journal of Computer Applications 178, no. 29 (2019): 13-15.

[18]Walczyk, C.J.; Moroz, L.V.; Cie´sli ´ nski, J.L. Improving the Accuracy of the Fast Inverse Square Root by ModifyingNewton–Raphson Corrections. Entropy 2021, 23, 86.

[19]Ukil, A.; Shah, V.H.; Deck, B. Fast computation of arctangent functions for embedded applications: A comparative analysis. In Proceedings of the 2011 IEEE International Symposium on Industrial Electronics, Gdansk, Poland, 27–30 June 2011; pp. 1206–1211.

[20] El-Kenawy, El-Sayed M., Marwa Eid, and Alshimaa H. Ismail. "A New Model for Measuring Customer Utility Trust in Online Auctions." International Journal of Computer Applications 975: 8887.

[21] Benammar, M.; Alassi, A.; Gastli, A.; Ben-Brahim, L.; Touati, F. New Fast Arctangent Approximation Algorithm for Generic Real-Time Embedded Applications. Sensors 2019, 19, 5148.

[22] Zhu, H.; Ge, Y.; Jiang, B. Modified CORDIC algorithm for computation of arctangent with variable iterations. In Proceedings of the 2016 IEEE 13th International Conference on Signal Processing (ICSP), Chengdu, China, 6–10 November 2016; pp. 261–264.

Doi : https://doi.org/10.54216/IJWAC.030205
Received: July 01, 2021 Accepted: August 30, 2021

[23] El-kenawy, El-Sayed M., Marwa M. Eid, and Abdelhameed Ibrahim. "Anemia estimation for covid-19 patients using a machine learning model." Journal of Computer Science and Information Systems 17, no. 11 (2021): 2535-1451.
[24] Pilato, L.; Fanucci, L.; Saponara, S. Real-Time and High-Accuracy Arctangent Computation Using CORDIC and Fast Magnitude Estimation. Electronics 2017, 6, 22.
[25] Ibrahim, Abdelhameed, Seyedali Mirjalili, Mohammed El-Said, Sherif SM Ghoneim, Mosleh M. Al-Harthi, Tarek F. Ibrahim, and El-Sayed M. El-Kenawy. "Wind speed ensemble forecasting based on deep learning using adaptive dynamic optimization algorithm." IEEE Access 9 (2021): 125787-125804.
[26] Eid, Marwa M., El-Sayed M. El-kenawy, and Abdelhameed Ibrahim. "A binary sine cosine-modified whale optimization algorithm for feature selection." In 2021 National Computing Colleges Conference (NCCC), pp. 1-6. IEEE, 2021.

Doi : https://doi.org/10.54216/IJWAC.030205
Received: July 01, 2021 Accepted: August 30, 2021