NEW COMPLEX VALUED ACTIVATION FUNCTIONS: COMPLEX MODIFIED SWISH, COMPLEX E-SWISH AND COMPLEX FLATTEN-T SWISH

Mehmet ÇELEBİ
Department of Project
İller Bankası A.Ş,
Ankara, Turkey

Murat CEYLAN
Department of Electrical and Electronics Engineering
Konya Technical University
Konya, Turkey

Abstract: Complex valued neural network (CVNN) has been developed to process complex valued data directly. In CVNN, one of the most important factors is selecting the node’s activation function. Choosing the right activation function for each layer is also crucial and may have a significant impact on metric scores and the training speed of the model. This paper introduces three new activation functions for CVNNs which is closely related to the activation function complex swish. These new activation functions are complex modified swish, complex E-swish and complex Flatten-T swish. In order to verify the validity and practicability of the proposed three new activation functions are tested and compared with complex swish activation function on complex valued four bit XOR problem, three inputs symmetry detection and the fading equalization problems. We show that complex E-swish ($\beta=1.4$) has the best overall performance when compared to other networks using complex swish, complex modified swish and complex Flatten-T swish activation functions on the considered tasks.

Keywords: Complex valued artificial neural network, Swish, Modified swish, E-swish, Flatten-T swish

I. INTRODUCTION

As an extension of real valued artificial neural networks (RVANN), complex valued artificial neural networks (CVANN) has been developed to process data with complex numbers directly without requiring any pretreatment. CVANN is one of the type of neural network consisting of complex numbers of parameters such as weights, threshold values, inputs and outputs [1].

CVANN are suitable for areas that deal with complex valued data such as image processing by taking the fourier transformation, radar, telecommunication and speech recognition. CVANN are also used in areas such as telecommunications and image processing can be found in the literature [2,3,4]. For example, the fading equalization problem has been successfully solved with a single complex-valued neuron with the highest generalization ability [5]. Also the fading equalization, symmetry detection and exclusive-or (XOR) problem can be successfully solved by a single complex valued neuron [5,6].

The threshold values and initial weights of CVANN, normalization of data and activation function of hidden nodes affects CVANN's convergence to target [7]. For CVANNs, the main task is finding an appropriate complex activation function. Despite the real valued activation function is usually selected to be bounded and smooth like a logarithmic sigmoid function, these properties are not suitable for CVANN [8].

All parameters of CVANN such as inputs, outputs and weights are in complex plane. So that the activation function of CVANN must be extended into the complex domain. The complex activation function should be satisfy the following features:
- The function $\phi(z)$ should not be linear in both the real and imaginary parts of $Z$, $ZR$ and $ZI$. Otherwise, the multi-layer perceptron will have no advantage. If correct, using a multi-layer perceptron would be equal to a single-layer perceptron [1].
- The function $\phi(z)$ must be bounded. The formulas described for the forward passage of the multilayered perceptron require limitation. Otherwise there will be interruptions during the training [1].
- The Partial derivatives of $\phi(z)$ should exist and be bounded. Since we use complex back-propagation, the partial derivatives of $\phi(z)$ need to be bounded [1].
- The function $\phi(z)$ must be defined as a complex function that is analytic all over the complex plane [1].

There are many studies in the field of complex activation function. Some of these are given below.

Leung and Haykin [9] was used the sigmoid function on complex domain, as follows:

$$f(z) = \frac{1}{1 + e^{-z}}$$

Leung and Haykin scaled the input data to some region of complex domain given below, because this function has singular points at every $z = (2n + 1)i\pi, n \in \mathbb{Z}$. 

$$f(z) = \frac{1}{1 + e^{-Rz}} + i \frac{1}{1 + e^{-Imz}}$$

Benvenuto and Piazza [10], Birx and Pipenberg [11] was adapted $\tanh$ function into CVANNs as real-imaginary type activation function given bellow:

$$f(z) = \tanh(Rez) + itanh(Imz)$$

Kechriotis and Monalakos [12], Kinouchi and Hagiwara [13] used $\frac{\tanh}{\tanh}$ function given below:
\[ f(z) = \tanh(|z|) \exp(i \arg(z)) \quad (4) \]

By Hirose [14].

\section*{II. MATERIALS AND METHOD}

\subsection*{A. Complex-Valued Neuron Model}

Complex-valued neurons (CVN) are as natural as complex numbers and they are more functional than real-valued neurons (RVN) such as learning faster and generalize better. CVN can be used for simulation of biological neurons and treat the phase information properly. A single RVN can learn only linearly-separable input/output mappings but CVN can learn both of them [15].

The complex valued neuron model is given in Eq.5 where the \( f \) is an activation function, \( x_1, \ldots, x_2 \) are the inputs, \( w_0, \ldots, w_n \) are the weights and \( P_0 \) is the activation function [15].

\[ f(x_1, \ldots, x_n) = P_0(w_0 + w_1 x_1 + \cdots + w_n x_n) \quad (5) \]

The activation function \( P_0 \) can be a real function \( P_0: C \rightarrow R \) or complex function \( P_0: R \rightarrow C \) but the function always acts on a complex variable [16].

One of the main advantages of CVN is their ability to work with the phase, which is very important for the analysis of signals and for solving different pattern recognition and classification problems. The analysis of real-valued signals one of the most efficient approaches is the frequency domain analysis, which immediately involves complex numbers. By analysing signal properties in the frequency domain, we can see that each signal is characterized by magnitude and phase that carry different information about the signal. So that CVN treat the phase information properly [15].

\subsection*{B. The New Activation Functions}

\textbf{Complex Swish}

Swish activation function was proposed by researchers at Google (2007). This activation function and its derivative are formulized by Eq. 6, 7 [17].

\[ y = x. \sigma(ax) \quad (9) \]

Where the \( a \) value is either a constant or a trainable parameter. Since the proposed CVANN will use the back propagation, the derivative of \( y \) is needed. The derivative of the \( y \) function is formulized by the following equations:

\[ y' = \sigma(ax) + ax. \sigma(ax). (1 - \sigma(ax)) \]

\[ y' = \sigma(ax) + ax. \sigma(ax) - ax. \sigma(ax)^2 \]

\[ y' = ax. \sigma(ax) + \sigma(ax)(1 - ax. \sigma(ax)) \]

\[ y' = ay + \sigma(ax)(1 - ay) \quad (10) \]

In our study, the modified swish activation function has been studied in complex plane with the complex valued input \( z \). Modified Swish has been adapted to CVANNs as real and imaginary type function as follows:

\[ y = \text{Modswish}(Re(z)) + i. \text{Modswish}(Im(z)) \quad (11) \]

\textbf{Complex E-Swish}

Alcaide introduced the E-swish activation function (2018) [18]. E-swish and its derivative are formulized as:

\[ y = \beta x. \sigma(x) \quad (12) \]

\[ y' = \beta \sigma(x) + \beta x. \sigma(x). (1 - \sigma(x)) \]

\[ y' = \beta \sigma(x) + \beta x. \sigma(x) - \beta x. \sigma(x)^2 \]

\[ y' = \beta x. \sigma(x) + \sigma(x) \beta - \beta x. \sigma(x)^2 \]

\[ y' = y + \sigma(x)(\beta - y) \quad (13) \]

E-swish activation function is very similar to Swish. Actually, when we take the \( \beta \) constant as 1, E-swish becomes the same as Swish. The constant value \( \beta \) is either a constant or a trainable parameter [18].

In our study, E-swish activation function has been studied in complex plane with the complex valued input \( z \). E-Swish has been adapted to CVANNs as real and imaginary type function given by:

\[ y = E\text{swish}(Re(z)) + i. E\text{swish}(Im(z)) \quad (14) \]

\textbf{Complex Flatten-T Swish}

This new activation function called FTS wish or Flatten-T Swish (FTS) was proposed by Chieng [19]. Flatten-T swish combines the Swish and Rectified Linear Units (ReLU) activations functions into a new one. FTS wish is formulized as follows:

\[ FTS(x) = \begin{cases} \frac{x}{1 + e^{-x}} + T, & x \geq 0 \\ T, & x < 0 \end{cases} \quad (15) \]

\( T \) is the parameters called threshold values that enable the negative part of the equation to produce negative values[19].
The proposed CVANN will use the back propagation algorithm so that the derivative of the formula is needed. $T$ is a constant value so that its derivative simply converts to be 0 (similarly, this applies to the FTS(x) derivative $x < 0$). So that, the only term in the derivative is $y = x \cdot \sigma(x) + T$. The derivative steps of FTS is given below:

$$y' = \sigma(x) + x \cdot \sigma(x)(1 - \sigma(x))$$
$$y' = \sigma(x) + x \cdot \sigma(x) - x \cdot \sigma(x)^2$$
$$y' = \sigma(x) + x \cdot y - \sigma(x) \cdot y$$
$$y' = \sigma(x)(1 - y) + y$$

(16)

As a whole, the derivative of the FTS is formulized as:

$$FTS(x) = \begin{cases} \sigma(x)(1 - y) + y, & x \geq 0 \\ 0, & x < 0 \end{cases}$$

(17)

In our study, FTS wish activation function has been studied in complex plane with the complex valued input z. FTSwish has been adapted to CVANNs as real and imaginary type function given by:

$$y = FTSwish(Re(z)) + i \cdot FTSwish(Im(z))$$

(18)

C. Complex Valued Data

Since all parameters in CVANN consist of complex numbers, the input data must also be in complex domain. Therefore the real valued input data must be moved to complex domain. In this study the real valued four bit XOR and three inputs one output symmetry detection problem converted to complex domain.

In this study, to verify the validity and applicability of the CVANN using the new activation functions, we applied it to three problems: the four bit XOR, three inputs one output symmetry detection and fading equalization problems.

Complex valued XOR problem with four patterns

The Complex valued Exclusive-Or (XOR) problem with four patterns is given in Table 1. The complex valued XOR problem is defined according to the following two rules:

- The real part of the output, the real and the imaginary part of the XOR
- The imaginary part of the output is taken as the real part of the input [20].

Table 1. Similar XOR for CVANN

| Input | Output |
|-------|--------|
| $X_1$ | $X_2$ | $X_3$ | $Y$ |
| 0     | 0     | 0     | 1   |
| i     | 1     | -i    | 0   |
| 1     | i+i   | i     | 0   |
| 1+i   | i     | 1     | 0   |

The CVANN has many advantages over the RVANN, such as the XOR problem, which can be solved with two layered CVANN [21].

Symmetry detection problem

The problem of symmetry determination aims to symmetrically determine the binary activity levels of a one-dimensional input neuron array under the central point. Symmetry probability decreases as the number of bits increases. Therefore, the symmetry detection problem is a very suitable problem for researching unbalanced data because the possibility of being symmetric decreases as the number of bits increases. Three inputs and one output detection of symmetry problem is shown in Table 2 [21].

Table 2. The detection of symmetry problem

| Inputs | Output |
|--------|--------|
| $X_1$ | $X_2$ | $X_3$ | $Y$ |
| 0     | 0     | 0     | 1   |
| 0     | 1     | 0     | 1   |
| 0     | 1     | 1     | 0   |
| 1     | 0     | 0     | 0   |
| 1     | 1     | 0     | 1   |
| 1     | 1     | 1     | 1   |

Since all parameters in CVANN consist of complex numbers, the input data must also be in complex domain. Therefore the real valued input data must be converted to complex domain. This conversion can be done with sample angle-based coding using the equation given below [22].

$$\varphi = \frac{\theta(x - a)}{b - a}$$

(19)

where $x \in [a,b]$ and $\theta$ the is the mapping angle. After the encoded angle value $\varphi$ is evaluated by a linear transformation using Eq. 19, by the Euler formula given in Eq. 20, the complex valued data is obtained on the unit circle with unity gain.

$$Z = e^{i\varphi} = \cos\varphi + i\sin\varphi$$

(20)

In this study, a and b were taken as 0 and 1, respectively. The data was moved to the complex plane with a phase angle $\theta = \pi/4$. Three input and one output symmetry detection problems in the complex plane are given Table 3.

Table 3. Complex valued symmetry detection

| Inputs | Output |
|--------|--------|
| $X_1$ | $X_2$ | $X_3$ | $Y$ |
| 1     | 1     | 0.7+0.7i | 1   |
| 1     | 0.7+0.7i | 1     | 0.7+0.7i |
| 1     | 0.7+0.7i | 0.7+0.7i | 1   |
| 0.7+0.7i | 1     | 1     | 0.7+0.7i |
| 0.7+0.7i | 1     | 0.7+0.7i | 0.7+0.7i |
| 0.7+0.7i | 0.7+0.7i | 1     | 0.7+0.7i | 0.7+0.7i |

© 2020-2022, IJARCS All Rights Reserved
**Fading equalization problem**

This section showed that the fading equalization problem can be successfully solved by the two-layered CVANN with the highest generalization ability [23]. Channel equalization problem in a digital communication system can be seen as the pattern classification problem. The digital communication system receives a signal sequence transmitted with additional noise and tries to estimate the actual transmitted sequence from these signals. A transmitted signal can assume one of the following four possible complex values: \(-1 - i , -1 + i , 1 - i \) and \(1 + i (i = \sqrt{-1}) \). Thus, the received signal will take value around \(-1 - i , -1 + i , 1 - i \) and \(1 + i \) because some noises are added. We need to estimate the true complex values from such complex values with noises. Thus, a method with excellent generalization ability is needed for the estimate. The input-output mapping in the problem is shown in Table 4 [24].

In order to solve the problem with the complex-valued neural network, the input-output mapping in Table 4 is encoded as shown in Table 5 [25].

**III. RESULTS**

**A. Complex-Valued XOR Problem with Four Patterns**

CVANN using the new activation functions (E-Swish, Flatten-T Swish, Modified Swish) was tested on complex valued XOR problem with four patterns for finding the best constant value(β, T and α). We use 1-2-1 network with the learning rate 0.5 as in the literature[5,28-32].The network was stopped when the error rate was achieved. For the error rate we use Mean squared error (MSE) value given below:

\[
E_p = MSE = (1/2) \sum_{n=1}^{N} |T_n - O_n|^2
\]  

(21)

In order to find the best β, T and α values, CVANN using E-Swish, Flatten-T Swish and Modified Swish activation functions was tested with 18 different values (randomly). When the error value (MSE) reached 0.001, the iteration number of the CVANNs are shown in Table 6,7,8. The numbers in “score” column report the aggregate number of times of E-swish, Flatten-T swish and modified swish activation functions gives the best result obtained by the existing activation function across the four tests (Test-I, Test-II, Test-III and Test-IV).

| Activation Function | Iteration number that reached the error rate | Score |
|---------------------|---------------------------------------------|-------|
| E-Swish (β = 0.1)   | 3000+ 3000+ 3000+ 3000+                   | 4     |
| E-Swish (β = 0.2)   | 2031 3000+ 3000+ 3000+                     | 3     |
| E-Swish (β = 0.3)   | 1218 3000+ 3000+ 3000+                     | 3     |
| E-Swish (β = 0.4)   | 930 2489 2344 23667                       | 3     |
| E-Swish (β = 0.5)   | 774 1756 1812 1888                        | 3     |
| E-Swish (β = 0.6)   | 699 1416 1494 1609                        | 3     |
| E-Swish (β = 0.7)   | 585 1201 1280 1419                        | 3     |
| E-Swish (β = 0.8)   | 512 1036 1163 1281                        | 3     |
| E-Swish (β = 0.9)   | 447 912 947 1176                         | 3     |
| E-Swish (β = 1.0)   | 391 846 846 1092                        | 3     |
| E-Swish (β = 1.1)   | 343 940 778 1023                        | 3     |
| E-Swish (β = 1.2)   | 304 1066 734 915                         | 3     |
| E-Swish (β = 1.3)   | 272 735 715 871                         | 3     |
| E-Swish (β = 1.4)   | 245 665 760 831                         | 4     |
| E-Swish (β = 1.5)   | 223 920 531 673                         | 3     |
| E-Swish (β = 2)     | 203 738 3000+ 475                         | 3     |
| E-Swish (β = 3)     | 100 792 3000+ 370                         | 3     |
| E-Swish (β = 4)     | 368 1000 3000+ 3000+                     | -     |

| Activation Function | Iteration number that reached the error rate | Score |
|---------------------|---------------------------------------------|-------|
| FTS (T = 1)         | 3000+ 3000+ 3000+ 3000+                   | 4     |
| FTS (T = 0.5)       | 3000+ 3000+ 3000+ 3000+                   | 4     |
| FTS (T = 0.3)       | 3000+ 3000+ 3000+ 3000+                   | 4     |
| FTS (T = 0.2)       | 1834 1005 1764 3000+                     | 4     |
| FTS (T = 0.1)       | 1558 3000+ 3000+ 3000+                   | 4     |
| FTS (T = 0)         | 3000+ 2222 3000+ 949                     | 4     |
| FTS (T = -0.1)      | 3000+ 575 3000+ 512                      | 2     |
| FTS (T = -0.2)      | 562 567 509 561                        | 3     |
| FTS (T = -0.3)      | 718 598 536 536                         | 4     |
| FTS (T = -0.4)      | 761 616 570 532                         | 3     |
| FTS (T = -0.5)      | 870 3000+ 532 532                       | 3     |
| FTS (T = -0.6)      | 1037 3000+ 556 626                      | 3     |
| FTS (T = -0.7)      | 3000+ 3000+ 575 2943                    | 3     |
| FTS (T = -0.8)      | 3000+ 3000+ 593 3000+                   | 3     |
| FTS (T = -0.9)      | 3000+ 3000+ 631 3000+                   | 3     |
| FTS (T = -1)        | 3000+ 3000+ 686 3000+                   | 3     |
| FTS (T = -2)        | 3000+ 3000+ 3000+ 3000+                 | -     |

| Activation Function | Iteration number that reached the error rate | Score |
|---------------------|---------------------------------------------|-------|
| Mod. Swish (α = 0.1)| 3000+ 3000+ 3000+ 3000+                   | 4     |
| Mod. Swish (α = 0.2)| 3000+ 3000+ 3000+ 3000+                   | 4     |
| Mod. Swish (α = 0.3)| 3000+ 3000+ 3000+ 3000+                   | 4     |
| Mod. Swish (α = 0.4)| 3000+ 3000+ 3000+ 3000+                   | 4     |
| Mod. Swish (α = 0.5)| 3000+ 3000+ 3000+ 3000+                   | 4     |
| Mod. Swish (α = 0.6)| 3000+ 3000+ 1794 896                     | 3     |

Rumelhart et al., (1986a, b) showed that increasing the number of layers raised the computational power of neural networks [26,27].
The results show that CVANN using E-Swish with $\beta=1.4$, Flatten-T Swish with $T=-0.3$ and Modified Swish with $\alpha=1.4$ converges to the target earlier than other CVANN using activation functions with different $\beta,T$ and $\alpha$ values on complex-valued XOR problem with four patterns.

### B. Complex-Valued Symmetry Detection Problem

For finding the best constant value for the new activation functions (E-Swish, Flatten-T Swish, Modified Swish), CVANN was tested on complex-valued symmetry detection problem with 18 different $\beta,T$ and $\alpha$ values (randomly). We use 3-1-1 (three input, one hidden nodes and one output) network with the learning rate 0.5 as in the literature [5,25]. We use MSE as stopping criteria given by Eq.21.

The iteration number when the proposed CVANNs error rate reached 0.001 is given in Table 9, 10, 11.

| Activation Function | Iteration number that reached the error rate 0.001(MSE) for symmetry problem |
|---------------------|-----------------------------------------------------------------------------|
| E-Swish ($\beta = 0.1$) | Test-I: 3000+ | Test-II: 3000+ | Test-III: 3000+ | Test-IV: 3000+ | Score |
| E-Swish ($\beta = 0.2$) | 2341 | 5703 | 1045 | 1191 | 3000+ |
| E-Swish ($\beta = 0.3$) | 2544 | 6253 | 1276 | 1321 | 3000+ |
| E-Swish ($\beta = 0.4$) | 2645 | 6423 | 1368 | 1422 | 3000+ |
| E-Swish ($\beta = 0.5$) | 2746 | 6624 | 1513 | 1555 | 3000+ |
| E-Swish ($\beta = 0.6$) | 2847 | 6825 | 1655 | 1707 | 3000+ |
| E-Swish ($\beta = 0.7$) | 2948 | 7026 | 1800 | 1852 | 3000+ |
| E-Swish ($\beta = 0.8$) | 3049 | 7227 | 1945 | 1997 | 3000+ |
| E-Swish ($\beta = 0.9$) | 3150 | 7428 | 2090 | 2142 | 3000+ |
| E-Swish ($\beta = 1.0$) | 3251 | 7629 | 2235 | 2287 | 3000+ |
| E-Swish ($\beta = 1.1$) | 3352 | 7830 | 2380 | 2432 | 3000+ |
| E-Swish ($\beta = 1.2$) | 3453 | 8031 | 2525 | 2577 | 3000+ |
| E-Swish ($\beta = 1.3$) | 3554 | 8232 | 2671 | 2722 | 3000+ |
| E-Swish ($\beta = 1.4$) | 3655 | 8433 | 2817 | 2869 | 3000+ |
| E-Swish ($\beta = 1.5$) | 3756 | 8634 | 2963 | 3015 | 3000+ |
| E-Swish ($\beta = 1.6$) | 3857 | 8835 | 3111 | 3163 | 3000+ |
| E-Swish ($\beta = 1.7$) | 3958 | 9036 | 3257 | 3310 | 3000+ |
| E-Swish ($\beta = 1.8$) | 4059 | 9237 | 3403 | 3456 | 3000+ |
| E-Swish ($\beta = 1.9$) | 4160 | 9438 | 3550 | 3603 | 3000+ |
| E-Swish ($\beta = 2.0$) | 4261 | 9639 | 3696 | 3750 | 3000+ |

| Activation Function | Iteration number that reached the error rate 0.001(MSE) for XOR problem |
|---------------------|-----------------------------------------------------------------------------|
| E-Swish ($\beta = 0.1$) | Test-I: 567 | Test-II: 1117 | Test-III: 768 | Test-IV: 723 |
| E-Swish ($\beta = 0.2$) | 270 | 674 | 414 | 448 |
| E-Swish ($\beta = 0.3$) | 174 | 526 | 340 | 344 |
| E-Swish ($\beta = 0.4$) | 128 | 500 | 320 | 242 |
| E-Swish ($\beta = 0.5$) | 102 | 519 | 301 | 164 |
| E-Swish ($\beta = 0.6$) | 85 | 468 | 271 | 117 |
| E-Swish ($\beta = 0.7$) | 74 | 442 | 237 | 88 |
| E-Swish ($\beta = 0.8$) | 65 | 425 | 207 | 71 |
| E-Swish ($\beta = 0.9$) | 59 | 412 | 181 | 59 |
| E-Swish ($\beta = 1.0$) | 54 | 401 | 161 | 50 |
| E-Swish ($\beta = 1.1$) | 50 | 391 | 144 | 35 |
| E-Swish ($\beta = 1.2$) | 47 | 382 | 129 | 33 |
| E-Swish ($\beta = 1.3$) | 44 | 374 | 117 | 32 |
| E-Swish ($\beta = 1.4$) | 42 | 365 | 107 | 26 |
| E-Swish ($\beta = 1.5$) | 39 | 356 | 98 | 36 |

It was seen that CVANN using E-Swish with $\beta=1.4$, Flatten-T Swish with $T=-0.2$ and Modified Swish with $\alpha=1.5$ converges to the target earlier than other CVANN using activation functions with different $\beta,T$ and $\alpha$ values on complex-valued symmetry detection problem.

### C. The Fading Equalization Problem

In the following text, it is shown that the fading equalization problem which cannot be solved with a single real-valued neuron, can be successfully solved by a single CVN. First we found the best constant value for the new activation functions (E-Swish, Flatten-T Swish, Modified Swish) on the fading equalization problem. We use a 1-2-1 CVANN with the learning constant 0.5. When the error value (MSE) reached 0.001, the iteration number of the CVANNs are shown in the Table 12, 13, 14.
Table 13. Convergence performance for complex FTS with vary of T values, shown in parenthesis, on fading equalization problem.

| Activation Function | Test-I (T = 1) | Test-II | Test-III | Test-IV | Score |
|---------------------|---------------|---------|----------|---------|-------|
| FTS (T = 0.5)       | 48            | 41      | 63       | 70      | 3     |
| FTS (T = 0.4)       | 50            | 42      | 69       | 71      | 3     |
| FTS (T = 0.3)       | 54            | 44      | 78       | 73      | 3     |
| FTS (T = 0.2)       | 41            | 46      | 97       | 75      | -     |
| FTS (T = 0.1)       | 77            | 50      | 149      | 78      | -     |
| FTS (T = 0)         | 169           | 56      | 237      | 82      | -     |
| FTS (T = -0.1)      | 54            | 273     | 121      | 99      | -     |
| FTS (T = -0.2)      | 57            | 322     | 254      | 183     | -     |
| FTS (T = -0.3)      | 61            | 194     | 245      | 196     | -     |
| FTS (T = -0.4)      | 65            | 162     | 206      | 163     | -     |
| FTS (T = -0.5)      | 65            | 136     | 169      | 134     | -     |
| FTS (T = -0.6)      | 63            | 112     | 136      | 110     | -     |
| FTS (T = -0.7)      | 61            | 96      | 112      | 91      | -     |
| FTS (T = -0.8)      | 60            | 86      | 94       | 76      | -     |
| FTS (T = -0.9)      | 64            | 83      | 81       | 64      | 1     |
| FTS (T = -1)        | 80            | 89      | 72       | 55      | 1     |
| FTS (T = -2)        | 97            | 1514    | 301      | 38      | 1     |

Table 14. Convergence performance for complex mod. swish with vary of α values, shown in parenthesis, on fading equalization problem.

| Activation Function | Test-I (α = 0.1) | Test-II | Test-III | Test-IV | Score |
|---------------------|------------------|---------|----------|---------|-------|
| Mod. Swish (α = 0.1) | 42               | 58      | 54       | 59      | 3     |
| Mod. Swish (α = 0.2) | 83               | 69      | 69       | 65      | 3     |
| Mod. Swish (α = 0.3) | 418              | 91      | 128      | 81      | 1     |
| Mod. Swish (α = 0.4) | 685              | 97      | 813      | 130     | -     |
| Mod. Swish (α = 0.5) | 456              | 120     | 438      | 654     | -     |
| Mod. Swish (α = 0.6) | 377              | 132     | 361      | 441     | -     |
| Mod. Swish (α = 0.7) | 348              | 115     | 328      | 365     | -     |
| Mod. Swish (α = 0.8) | 341              | 98      | 312      | 336     | -     |
| Mod. Swish (α = 0.9) | 340              | 86      | 312      | 327     | -     |
| Mod. Swish (α = 1.0) | 339              | 77      | 314      | 324     | -     |
| Mod. Swish (α = 1.1) | 335              | 70      | 315      | 322     | -     |
| Mod. Swish (α = 1.2) | 332              | 65      | 316      | 319     | -     |
| Mod. Swish (α = 1.3) | 328              | 60      | 317      | 329     | -     |
| Mod. Swish (α = 1.4) | 324              | 56      | 319      | 192     | -     |
| Mod. Swish (α = 1.5) | 321              | 53      | 322      | 204     | 1     |
| Mod. Swish (α = 2)  | 299              | 43      | 332      | 326     | 2     |
| Mod. Swish (α = 3)  | 313              | 32      | 302      | 61      | 2     |
| Mod. Swish (α = 4)  | 304              | 27      | 277      | 41      | 4     |

As seen in the results, the CVANN using E-Swish with β = 1.4, Flatten-T Swish with T = 1 and Modified Swish with α ≈ 4 gives the best results compared with the other β, T and α values on the fading equalization problem.

D. Comparison of Four Activation Functions

In order to prove the validity of proposed CVANNs, three new activation functions (modified swish with a constant value α, complex E-swish with a constant value β and complex Flatten-T swish with a constant value T) are tested and compared with complex swish activation function on complex valued four bit XOR problem, three inputs symmetry detection and the fading equalization problems. When the error value (MSE) reached 0.001, the average learning epochs are shown in the Table 15, 16, 17.

The experiments show that the proposed CVANN using complex E-Swish activation function with β=1.4 has better stability convergence performance than the other complex activation functions on complex valued XOR, symmetry and fading equalization problems. The average learning epochs, targets and outputs of Test-V are given in the Table 18-20 when the error value (MSE) reached the stopping criteria 0.001.

Table 15. Convergence performance of swish, E-swish, Flatten-T swish and modified swish on XOR problem.

| Activation Function | Iteration number that reached the error rate 0.001(MSE) for XOR problem |
|---------------------|-----------------------------|
| Swish               | 592                         |
| E-Swish (β = 1.4)   | 266                         |
| F.T Swish (T = 0.3) | 499                         |
| Mod. Swish (α = 1.4)| 1045                        |

Table 16. Convergence performance of swish, E-swish, Flatten-T swish and modified swish on symmetry problem.

| Activation Function | Iteration number that reached the error rate 0.001(MSE) for Symmetry problem |
|---------------------|-----------------------------|
| Swish               | 215                         |
| E-Swish (β = 1.4)   | 174                         |
| F.T Swish (T = 1)   | 239                         |
| Mod. Swish (α = 1.5)| 193                         |

Table 17. Convergence performance of swish, E-swish, Flatten-T swish and modified swish on fading equalization problem.

| Activation Function | Iteration number that reached the error rate 0.001(MSE) on fading problem |
|---------------------|-----------------------------|
| Swish               | 88                          |
| E-Swish (β = 1.4)   | 56                          |
| F.T Swish (T = 1)   | 56                          |
| Mod. Swish (α = 4)  | 74                          |

Table 18. The New CVANN Test Results for XOR Problem (Test-V)

| Activation Function | Iteration number with an error rate of 0.001: |
|---------------------|-----------------------------|
| E-SWISH             | 451                         |
| MOD-SWISH           | 625                         |
| FT-SWISH            | 651                         |

Table 19. The new CVANN test results for symmetry problem (Test-V)

| Activation Function | Iteration number with an error rate of 0.001: |
|---------------------|-----------------------------|
| E-SWISH             | 212                         |
| MOD-SWISH           | 231                         |
| FT-SWISH            | 327                         |

© 2020-2022, IJARCS All Rights Reserved
It is also showed that t
earlier.

Complex E-swish with a constant value  
Flatten-T swish with a constant value  
Swish with \( \beta = 1.4 \), complex modified swish with \( \beta = 1.4 \), complex modified swish with  

called complex modified swish with a constant value  

On XOR problem. According to symmetry problem tests, it  

Our experiments has shown that complex E-swish with  
complex Flatten-T swish with a constant value \( T \) have been presented. It is also showed that the parameters \( \beta, T \) and \( \alpha \) determine the convergence of CVANNs to the target and the training speed of the model.

Our experiments has shown that complex E-swish with  
complex Flatten-T swish with \( T = -0.3 \) converges to the target earlier on XOR problem. According to symmetry problem tests, it has shown that complex E-swish with \( \beta = 1.4 \), complex modified swish with \( \alpha = 1.5 \) and complex Flatten-T swish with \( T = -0.2 \) converges to the target earlier than the other values. Finally, the CVANN is tested on the fading equalization problem. The results showed that complex E-swish with \( \beta = 1.4 \), complex modified swish with \( \alpha = 4 \) and complex Flatten-T swish with \( T = 1 \) converges to the target earlier.

The performance of the three new activation functions with the best constant value, is compared with the new swish activation function on complex XOR, symmetry and the fading equalization problems. After training the activation functions on this benchmark tests, all results show that E-Swish with \( \beta = 1.4 \) and \( \beta = 1.5 \) has the best achievement (noted with an asterisk*) among the existing activation functions in term of iteration number that reached the error rate. The Means Squared Error was used as a performance index.

From the presented experiments, apparently, we conclude that CVANN using complex E-swish activation function with \( \beta = 1.4 \) has the best overall performance when compared to other networks using complex swish, complex modified swish and complex Flatten-T swish activation functions on complex valued four bit XOR problem, three inputs symmetry detection and the fading equalization problems.

| Activation Function: | Activation Function: | Activation Function: |
|----------------------|----------------------|----------------------|
| E-SWISH              | MOD-SWISH            | FT-SWISH             |
| Iteration number with an error rate of 0.001: | Iteration number with an error rate of 0.001: | Iteration number with an error rate of 0.001: |
| 26                   | 31                   | 39                   |
| Target | Output | Target | Output | Target | Output |
| 0 0.024+0.060i | 0 0.043+0.043i | 0 0.029+0.049i |
| 0 0.016+0.996i | 0 0.016+0.986i | 0 0.022+0.991i |
| 0 0.989+0.047i | 0 0.980+0.030i | 0 0.982+0.024i |
| 1+i 0.979+0.985i | 1+i 0.967+0.963i | 1+i 0.972+0.968i |

**IV. DISCUSSION AND CONCLUSION**

In this paper, three new activation functions which are called complex modified swish with a constant value \( \alpha \), complex E-swish with a constant value \( \beta \) and complex Flatten-T swish with a constant value \( T \) have been presented. It is also showed that the parameters \( \beta, T \) and \( \alpha \) determine the convergence of CVANNs to the target and the training speed of the model.

Our experiments has shown that complex E-swish with \( \beta = 1.4 \), complex modified swish with \( \alpha = 1.4 \) and complex Flatten-T swish with \( T = -0.3 \) converges to the target earlier on XOR problem. According to symmetry problem tests, it has shown that complex E-swish with \( \beta = 1.4 \), complex modified swish with \( \alpha = 1.5 \) and complex Flatten-T swish with \( T = -0.2 \) converges to the target earlier than the other values. Finally, the CVANN is tested on the fading equalization problem. The results showed that complex E-swish with \( \beta = 1.4 \), complex modified swish with \( \alpha = 4 \) and complex Flatten-T swish with \( T = 1 \) converges to the target earlier.

The performance of the three new activation functions with the best constant value, is compared with the new swish activation function on complex XOR, symmetry and the fading equalization problems. After training the activation functions on this benchmark tests, all results show that E-Swish with \( \beta = 1.4 \) and \( \beta = 1.5 \) has the best achievement (noted with an asterisk*) among the existing activation functions in term of iteration number that reached the error rate. The Means Squared Error was used as a performance index.

From the presented experiments, apparently, we conclude that CVANN using complex E-swish activation function with \( \beta = 1.4 \) has the best overall performance when compared to other networks using complex swish, complex modified swish and complex Flatten-T swish activation functions on complex valued four bit XOR problem, three inputs symmetry detection and the fading equalization problems.

**V. REFERENCES**

[1] M. Ceylan, "Improving an algorithm with complex-valued artificial neural network and applications," MS Thesis, Selcuk University Graduate School of Natural and Applied Sciences, Konya, 13-14, 2004.

[2] A. Hirose, “Complex-Valued Neural Networks, World Scientific Publishing,” Singapore, 2003.

[3] A. Hirose, “Complex-Valued Neural Networks, Series on Studies in Computational Intelligence,” Vol. 32, Springer-Verlag, New York 2006.

[4] T. Nitta, “Complex-Valued Neural Networks: Utilizing High-Dimensional Parameters,” Information Science Reference, Hershey, PA, 2009.

[5] T. Nitta, “Solving the XOR problem and the detection of symmetry using a single complex valued neuron,” Neural Networks, 16:1101-1105, 2003.

[6] T. Nitta, “Orthogonality of Decision Boundaries in Complex-Valued Neural Networks,” Neural Computation 16, 73–97, 2004.

[7] M. Celebi, M. Ceylan, “The New Activation Function for Complex Valued Neural Networks: Complex Swish Function,” ISAS WINTER-Samsun, 2019.

[8] N. Özdemir, B. Iskender, N. Özgüür, “Complex valued neural network with Möbius activation function,” Commun Nonlinear Sci Numer Simulat 16, 2011.

[9] H. Leung, S. Haykin, “The complex backpropagation algorithm,” IEEE Trans Signal Process,39:2101–4, 1991.

[10] N. Benvenuto, F. Piazza, “On the complex backpropagation algorithm,” IEEE Trans Signal Process 40:967–9, 1992.

[11] D.L. Birx, S.J. Pipenberg, “Chaotic oscillators and complex mapping feed forward networks (CMFFNS) for signal detection in noisy environments,” Proceedings of IEEE international j. conf. neural networks. II, p. 881–888, 1992.

[12] G. Kechriotis, E.S. Monalakos, “Training fully recurrent neural networks with complex weights,” IEEE Trans Circuits Syst II, 41:235–8, 1994.

[13] M. Kinouchi, M. Hagiwara, “Learning temporal sequences by complex neurons with local feedback,” Proceedings of IEEE international conference on neural networks IV, p. 3165–3169,1995.

[14] A. Hirose, “Applications of complex-valued neural networks to coherent optical computing using phase-sensitive detection scheme,” Inf Sci Appl, 2:103–17, 1994.

[15] I. Aizenberg, “Complex-Valued Neural Networks with Multi-Valued Neurons, International Joint Conference on Neural Networks,” IJCNN, 2013.

[16] N. Moning, S. Manandhar, “Evaluation of Complex-Valued Neural Networks on Real-Valued Classification Tasks,” CoRR abs/1811.12351, 2018.

[17] P. Ramachandran, B. Zoph, “Searching for activation functions,” In International Conference on Learning Representations Q. V. L., 2018.

[18] E. Alcaide, “E-swish: Adjusting activations to different network depths,” arXiv preprint arXiv:1801.07145, 2018.

[19] H. Chieng, N. Wahid, O. Pauline, S. Perla, “Flatten-T Swish: a thresholded ReLU-Swish-like activation
function for deep learning,” International Journal of Advances in Intelligent Informatics, 2018.

[20] M. Ceylan, “Combined complex-valued artificial neural network (CCVANN),” Proceedings of the World Congress on Engineering. 2:955-959, 2011.

[21] X. Chen, “An Modified Error Function for the Complex-value Backpropagation Neural Networks,” Neural Information Processing - Letters and Reviews Vol.8, No.1, 2005.

[22] Y. Acar, M. Ceylan, E. Yaldız, “An examination on the effect of CVNN parameters while classifying the real-valued balanced and unbalanced data, International Conference on Artificial Intelligence and Data Processing (IDAP),” 2018.

[23] T. Nitta, “Orthogonality of Decision Boundaries in Complex-Valued Neural Networks,” Neural Computation 16, 73–97, 2004.

[24] T. Nitta, “Complex-Valued Neural Network and Complex-Valued Backpropagation Learning Algorithm,” Advances in Imaging and Electron Physics, Volume 152, ISSN 1076-5670, 2008.

[25] T. Nitta, “The Computational Power of Complex-Valued Neuron,” ICANN/ICONIP, 2003.

[26] D.E. Rumelhart, G.E. Hinton and R.J. Williams, “Learning internal representations by error propagation, Parallel Distributed Processing: Explorations in the microstructures of cognition,” vol. 1, pp. 318–362. Cambridge, MA: MIT Press, 1986.

[27] D.E. Rumelhart, G.E. Hinton and R.J. Williams, “Learning representations by back-propagating errors,” Nature 323, 533–536, 1986.

[28] H.A. Jalab, R. W. Ibrahim, “New activation functions for complex-valued neural network,” International Journal of the Physical Sciences Vol. 6(7), pp. 1766-1772, 4 April, 2011.

[29] T. Enomoto, K. Kakuda, S. Miura, “New Activation Functions in CNN and Its Applications,” ICCES, vol.1, no.2, pp.36-39, 2019.

[30] M. Ceylan, “Combined complex-valued artificial neural network (CCVANN),” Proceedings of the World Congress on Engineering. 2:955-959, 2011.

[31] H. Gürüler, M. Peker, “A Software Tool for Complex-Valued Neural Network: CV-ANN,” 23nd Signal Processing and Communications Applications Conference (SIU), 2015.

[32] X. Chen, Z. Tang C. Variappan, S. Li and T. Okada, “A modified error backpropagation algorithm for complex-value neural networks,” International Journal of Neural Systems, 15, 435-443, 2005.