ICASSP 2023 - 2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) | 978-1-7281-6327-7/23/$31.00 ©2023 IEEE | DOI: 10.1109/ICASSP49357.2023.10095457

# LEARNING PROPERTIES OF HOLOMORPHIC NEURAL NETWORKS OF DUAL VARIABLES

*Dmitry Kozlov*[†] Mikhail Bakulin[†] Stanislav Pavlov[†‡*] Aleksandr Zuev[†]
Mariya Krylova[†‡]
Igor Kharchikov[†‡]

† NN AI Team, Huawei Russian Research Institute, Nizhny Novgorod, Russia
‡ National Research University Higher School of Economics, Nizhny Novgorod, Russia
* Volga State University of Water Transport, Nizhny Novgorod, Russia

## ABSTRACT

Artificial neural networks have become an inseparable element of human life. Researches do not stop at the current progress and try to improve neural networks and expand fields of applications. The most widespread way to make models better consists in generalization of existing methods and approaches. In this paper, we make a step in an unusual direction: we propose to use neural networks based on dual numbers. We develop a special subclass of dual-valued operators, which satisfy the equivalent of the Cauchy-Riemann equations for the dual domain. We also propose a new type of preprocessing and batch normalization, relying on peculiarities of dual numbers. We test deep holomorphic dual-valued models on music transcription and gravitational wave detection tasks and show that our holomorphic dual-valued networks achieve better inference time compared to the dual-valued models and are better than their real-valued counterparts in sense of metrics.

***Index Terms***— deep learning, dual-valued neural networks, dual-valued batch normalization, dual number, dual modulus, holomorphic dual-valued operators, holomorphic dual-valued neural networks

## 1. INTRODUCTION

Neural networks have proven themselves as a powerful tool for various types of tasks, such as computer vision, object tracking, signal processing, natural language processing, and many others. Nowadays, the majority of researches focus on improvement of metrics, inference time and convergence of deep learning models. There are multiple strategies of pursuing these goals. The most popular are: usage of data augmentation, sophisticated optimizers/schedulers and changing architecture of neural networks. In some works, it has been noted that input of neural networks is not limited to the real-valued format. There are multiple tasks, where the original data have complex-valued representations. For example, wind observation, where wind speed and its direction are simultaneously measured, so data can be stored as complex numbers [1]. It is also a common approach that time series of real-valued data of any source are converted into the complex domain by Fourier transform.

Many researchers have tried to extend real-valued neural networks to the complex domain and have showed multiple benefits of using neural networks based on complex numbers. For example, complex-valued networks have larger representational capacity [2] and higher convergence rate [3], comparing to real models. Moreover, equivariant and invariant layers for neural networks based on complex numbers, presented in recent articles [4, 5], provide developers with new operators that help to build more robust models with better accuracy and generalization ability.

Generally, we suppose that the efficiency of complex models is achieved by taking into account phase information of the signal, because the real and imaginary parts of any layer's output can interact with each other. Thus, we do not discard information and operate with it in an appropriate way.

A survey [6] of existing complex-valued networks and developed approaches also shows growing interest in this field of artificial intelligence. These promising results urge us to make a step further and generalize neural networks to the dual domain $\mathbb{D}$. Dual numbers were introduced in 1873 by English geometer William Clifford (1845-1879) and were exploited by the German mathematician Eduard Study (1862-1930) to represent the dual angle which measures relative position of two skew lines in space [7, 8].

The dual numbers are a special kind of two-component numbers, whose elements have a form $x + \varepsilon y$, where $x, y$ are real numbers, and $\varepsilon$ is a nilpotent element, which satisfies the relations: $\varepsilon^2 = 0, \varepsilon \neq 0$. Similarly to the complex numbers, the real number $Re(z) = x$ will be called the real part of the dual number $z$, and the real number $Im(z) = y$ will be called the imaginary part of the dual number $z$.

As well as complex numbers, dual numbers have found applications in fundamental sciences such as quantum mechanics [9], Screw theory [10], and Riemannian geometry [11]. Dual numbers also make it possible to automatically compute derivatives of functions [12, 13]. It appears that the first attempt to develop neural network based on dual variables is [14], but this work does not exploit any properties of dual numbers, except for $\varepsilon^2 = 0$. In our previous work [15], we adjusted basic layers such as Linear, Convolution, Average Pooling, ReLU to the dual domain and presented an algorithm for Dual Batch Normalization.

In this paper, we exploit the Cauchy-Riemann conditions for functions of dual variable and introduce holomorphic dual-valued operators for deep learning models. In addition, we leverage matrix representation of dual numbers to generalize batch normalization technique [16] for the dual domain.

In section 2 of this paper, we define the norm of dual numbers, clarify peculiarities of the Cauchy-Riemann conditions for the dual-valued function, and provide information about usage of dual numbers in automatic differentiation.

Section 3 contains description of developed non-holomorphic and holomorphic dual-valued building blocks for our deep learning models. A special subsection is dedicated to data preprocessing that is needed to convert a real-valued input to the dual domain.

In section 4 of this paper, we show that holomorphic dual-valued neural networks is a reasonable balance between growth in computational complexity and achievement of higher metrics.

## 2. PROPERTIES OF DUAL NUMBERS

In this section, we provide more advanced information about some aspects of dual numbers, especially, we focus on derivative of functions of dual variables.

### 2.1. Fundamentals of dual numbers

The basic operations for dual numbers $x + \varepsilon y$ are

$$\begin{gathered}(x_1 + \varepsilon y_1) \pm (x_2 + \varepsilon y_2) = (x_1 \pm x_2) + \varepsilon(y_1 \pm y_2),\\ (x_1 + \varepsilon y_1)(x_2 + \varepsilon y_2) = x_1 x_2 + \varepsilon(x_1 y_2 + y_1 x_2),\\ \frac{(x_1 + \varepsilon y_1)}{(x_2 + \varepsilon y_2)} = \frac{x_1}{x_2} + \varepsilon \frac{(y_1 x_2 - x_1 y_2)}{x_2^2},\\ \overline{(x + \varepsilon y)} = (x - \varepsilon y).\end{gathered}$$

We can also consider a square of the dual modulus $x + \varepsilon y$ as

$$|z|^2 = z\overline{z} = (x + \varepsilon y)\overline{(x + \varepsilon y)} = x^2, \tag{1}$$

but this definition (1) does not depend on the imaginary part $y$. To develop a new definition of dual number modulus, we rely on their matrix representation. It turns out that any manipulations with dual numbers $x + \varepsilon y$ can be replaced by operations on a $2 \times 2$ matrix of the form $\begin{pmatrix} x & y \\ 0 & x \end{pmatrix}$.

We exploit this isomorphism and define the dual modulus as the subordinate norm of the corresponding matrix in the vector space $K^{2\times 2}$. The final equation for the dual modulus is:

$$\|z\| = \left|\frac{y}{2}\right| + \sqrt{x^2 + \left(\frac{y}{2}\right)^2}. \tag{2}$$

Equation (2) is essential for batch normalization.

### 2.2. Cauchy-Riemann Conditions for Function of Dual Variables

Originally, the Cauchy–Riemann equations are certain criteria needed for a complex function $f(x+iy) = u(x,y)+iv(x,y)$ to be holomorphic (complex differentiable), where $u$ and $v$ are real-valued functions of two variables. These equations impose restrictions for $u(x, y)$ and $v(x, y)$:

$$\frac{\partial u}{\partial x} = \frac{\partial v}{\partial y}, \qquad \frac{\partial u}{\partial y} = -\frac{\partial v}{\partial x}. \tag{3}$$

There are analogous conditions for a dual-valued function $f(x + \varepsilon y) = u(x, y) + \varepsilon v(x, y)$ to be holomorphic (in sense of dual numbers):

$$\frac{\partial u}{\partial x} = \frac{\partial v}{\partial y}, \qquad \frac{\partial u}{\partial y} = 0. \tag{4}$$

Using Taylor series expansion for dual-valued step, one can show that the above restrictions imply that a holomorphic function of the dual variable is expanded to the following form:

$$f(x + \varepsilon y) = f(x) + \varepsilon y f'(x). \tag{5}$$

Property (5) is called automatic differentiation, mentioned in Introduction. It means that, in order to calculate a derivative of the function $f$ at the point $x$, we just need to find its value of$x+\varepsilon$ and take the imaginary part of the result $Im(f(x+\varepsilon))$. In this paper, we consider dual functions and operators the analytic continuation of real ones. Therefore, we assume that $f(x)$ is real for any real $x$. To be clear, this is a sufficient but not a necessary condition for a function to be holomorphic.

## 3. METHODOLOGY

In this section we define main operations needed for neural networks: linear layer and batch normalization, which satisfy the Cauchy–Riemann conditions (4) and are derived using the proposed dual modulus (2). We also describe the procedure for dual-valued input generation.

### 3.1. Data Representation in the Hypercomplex Algebra

Although operations on complex numbers are integrated into modern frameworks such as PyTorch, dual numbers are not supported by any framework. In this work, we represent complex and dual tensors as two-channel real-valued tensors. We use tensors of this structure as arguments. Complex- and dual-valued layers return values in the same form. The peculiarities of every type of algebras are taken into consideration by a specific realization of network operators. We also emphasize that, thanks to the real-valued representation, we do not need to implement any complex- or dual-valued gradient.

### 3.2. Convolution

Convolution of an input is needed to extract a feature map and pass the output to the next layer. The formula for a $2D$ real-valued convolution is given below:

$$(X*A+b)_{i,j} = \sum_{n=0}^{N}\sum_{m=0}^{M} X_{i+n,j+m}A_{N-n,M-m}+b_{i,j}, \quad (6)$$

where $X$, $A$, $b$ are respectively the input, weights and bias (shift), $*$ is a convolution mark, $N \times M$ is a kernel size. In the case of complex-valued neural networks, i.e. when weights compose a complex matrix $A = A_r + iA_i$ and the input and bias are also complex-valued $Z = X + iY$ and $b = b_r + ib_i$, complex-valued convolution is expressed through four real-valued convolutions:

$$Z*A+b = (X*A_r - Y*A_i + b_r) + i(Y*A_r + X*A_i + b_i), \quad (7)$$

because of distributive properties, as mentioned in [17]. The extension of (7) to the dual domain ($i \Rightarrow \varepsilon$) is

$$Z * A + b = X * A_r + b_r + \varepsilon(Y * A_r + X * A_i + b_i). \quad (8)$$

It is easy to notice that (8) has three real-valued convolution, which leads to the theoretical 25% inference speed-up, compared to the complex convolution for the same number of parameters. One can see, that formula (8) in general case of weight matrix $A$ does not satisfy equations (4). To make sure a dual convolution is holomorphic, we must impose a restriction $Im(A) \equiv 0$. This condition is based on the fact that, for a linear function $f(x + \varepsilon y) = a_r x + b_r + \varepsilon(a_r y + a_i x + b_i)$ the limits of its increment, as the argument approaches zero along the real axis or the imaginary axis, are equal if and only if the condition $a_i = 0$ is true. These limits are:

$$\lim_{\Delta x \to 0} \frac{f(x + \Delta x + \varepsilon y) - f(x + \varepsilon y)}{\Delta x} = a_r + \varepsilon a_i \quad (9)$$

$$\lim_{\varepsilon \Delta y \to 0} \frac{f(x + \Delta x + \varepsilon(y + \Delta y)) - f(x + \varepsilon y)}{\varepsilon \Delta y} = a_r. \quad (10)$$

Thus, we define a dual holomorphic convolution, which satisfies (4) as following:

$$Z * A + b = X * A_r + b_r + \varepsilon(Y * A_r + b_i). \quad (11)$$

The definition (11) implies that only two real-valued convolution are needed. In other words, dual holomorphic convolution is potentially 2 and 1.5 times faster than the corresponding complex and dual operators. We should emphasize that conditions (4) do not affect bias $b$, and it is still dual-valued.

### 3.3. Batch Normalization

According to [16], batch normalization has the following definition:

$$\check{x} = \gamma \frac{x - E[x]}{\sqrt{Var[x] + \delta}} + \beta, \quad (12)$$

where $x$, $E[x]$, $Var[x]$ are the input, its mean and standard deviation, $\delta = 10^{-5}$ is needed to avoid division by zero. $\gamma$ and $\beta$ are trainable scale and shift. The standard deviation $Var[x]$ is a measure the difference between the value of an observation and the mean of the population $E[x]$. To generalize (12) to the non-real case, we propose to calculate distance (modulus) between dual-valued input and its mean value via formula (2). In the case of complex neural networks, we use the standard formula of complex number modulus $\sqrt{x^2 + y^2}$. In addition, for generalized batch normalization $\gamma$ and $\beta$ are two-component dual or complex numbers.

It should be noted that, in order to make this layer holomorphic, we just need to set $Im(\gamma) \equiv 0$, because division by $\sqrt{Var[x]}$ is real-valued.

### 3.4. Generation of Dual-valued Input

Just like complex ones, dual numbers are essentially pairs of real values. Basing on this similarity, we propose to use the complex-valued input in both dual and complex neural networks. In this paper, we use two techniques to convert real input to the complex format: Fourier transform for the MusicNet dataset and constant-Q transform ($CQT$) for the G2Net dataset.

We also develop an alternative variant of transformation based on equation (5). We notice that in (5) $Im(f(x + \varepsilon y))$ is mainly determined by the derivative of the function at the same point as $Re(f(x + \varepsilon y)) = f(x)$. Basing on this, we propose to transform real-valued numbers of input to the dual numbers as follows:

$$Input \Rightarrow Input + \varepsilon(Input)', \quad (13)$$

where $(Input)'$ is a function of $Input$, which in a sense we call a derivative of that input. The specific definition of the derivative depends on the task. For example, if input is a time series then it seems natural to define the derivative with respect to time as the difference of signal strength at adjacent time points.

## 4. EXPERIMENTAL RESULTS

To check our hypothesis, we carry out several experiments on classification problems: gravitational wave detection and music transcription task.

### 4.1. Gravitational Wave Detection

This dataset consists of simulated noised signals similar to ones recorded by a system of three ground-based laser interferometers: LIGO Hanford, LIGO Livingston, and Virgo. These signals are generated during cosmic events such as black holes absorbing neutron stars. We leverage $CQT$ algorithm as preprocessing, because it is supposed to be efficient for analysis of gravitational waves [18]. It transforms time series into a frequency map consisting of real and imaginary parts. While complex and dual-valued networks take both parts, input of the holomorphic dual-valued network is formed differently. The modulus of $CQT$ frequency portrait is used as a real component, and its derivative is used as an imaginary part (13). For this task we used Sobel operator[19] as discrete differentiation analogue. To classify whether a wave is present at the spectrogram we use ResNet18 [20], where all the layers are replaced by the corresponding complex/dual/holomorphic operators. Also before fully-connected layer net has 512 channels for both real and imaginary parts, to handle all information standard linear layer is replaced by sequentially connected concatenation operator and linear with two times more input channels (1024). In terms of memory efficiency, holomorphic dual-valued networks are about the same size as real-valued ones, while dual and complex models are two times larger.

**Table 1**. Average precision on the G2Net dataset.

| Model | Input | BN | AP, % |
|---|---|---|---|
| Real | $\|CQT\|$ | Real | 76.5 |
| Complex | $CQT$ | Complex | 78.7 |
| Dual | $CQT$ | Complex | 73.5 |
| Dual | $CQT$ | Dual | 79.2 |
| Dual | $\|CQT\| + \varepsilon\|CQT\|'$ | Dual | 51.73 |
| HDual | $CQT$ | Dual | 77.0 |
| HDual | $\|CQT\| + \varepsilon\|CQT\|'$ | Dual | 77.6 |
| HDual | $\|CQT\| + \varepsilon\|CQT\|'$ | HDual | 78.4 |

From the Table 1 it is clear that proper combination of batch normalization operator alongside preprocessing show the best results, furthermore all of these combinations outperform real-valued network.

### 4.2. Music Transcription Task

In this part we show the results of automatic music transcription. The goal of this task is classification of the notes in music recordings. The experiments are performed on the MusicNet dataset [21]. We adopt the DeepConvNet architecture developed in [17] and use frequency representation of the data, which we get after Fast Fourier transform (FFT). For the real neural networks, we consider the real and imaginary components of the data as separate channels. In the case of dual and complex-valued models, we use output of FFT without any changes. We use equation (13) to generate data for holomorphic dual neural networks. To do that we assign the real part to absolute value of the output of FFT and the imaginary component to derivative of the this absolute value, which is calculated via finite deference. This derivative reflects changes in intensity of the spectrum.

For each type of model's algebra we use respective neural operators, which were described in the Methodology section. The results are summarized in Table 2.

**Table 2**. Results of experiments on the MusicNet dataset.

| Model | Average Precision, % | Inference time, ms |
|---|---|---|
| Real | 68.9 | 0.82 |
| Complex | 72.5 | 3.56 |
| Dual | 73.0 | 3.16 |
| HDual | 71.2 | 2.86 |

From Table 2, we can observe that the best average precision is achieved by the dual-valued neural network. Meanwhile, the precision of the complex-valued model is close to the value reported in [17]. These results show that a dual-valued neural network has the best accuracy-performance trade-off, because it is faster than a complex-valued model and more accurate than other models. We consider dual-valued neural networks to be promising solutions for the tasks with the complex representation of input data.

## 5. CONCLUSION

Neural networks based on complex numbers tend to become a modern solution of tasks with complex-valued data. As discussed in this paper, models on dual numbers algebra deserve just as much attention of researchers. These models are capable to reach better metrics than their real-valued equivalents. As complex- and dual-valued models gain popularity, the question of balancing efficiency and performance requires greater efforts. Essentially, choosing type of algebra is a two-criteria problem. Holomorphic dual models offer a reasonable solution. As other networks on the second order algebras, they show better metrics than corresponding real-valued models, being just a little behind the dual models. At the same time they offer a high theoretical performance improvement. In practice inference speed-up depends on the architecture and is 10-25% compared to dual-valued models, which may be worth a trade-off of some accuracy. In addition, holomorphic models have about half as many parameters as dual models of the same architecture. These advantages make holomorphic dual networks a viable option in case of hardware limitations.

## 6. REFERENCES


[1] S.L. Goh et al., "Complex-valued forecasting of wind profile," *Renewable Energy*, vol. 31, no. 11, pp. 1733–1750, 2006.

[2] Tohru Nitta, "The computational power of complex-valued neuron," in *ICANN/ ICONIP 2003*, 2003, pp. 993–1000.

[3] Martin Arjovsky, Amar Shah, and Yoshua Bengio, "Unitary evolution recurrent neural networks," 2015.

[4] R. Chakraborty, Y. Xing, and S. X. Yu, "Surreal: Complex-valued learning as principled transformations on a scaling and rotation manifold," *IEEE Trans. Neural Networks Learn. Syst.*, vol. 33, no. 3, pp. 940—-951, 2020.

[5] U. Singhal, Y. Xing, and S. X. Yu, "Co-domain symmetry for complex-valued deep learning," 2021.

[6] Joshua Bassey, Lijun Qian, and Xianfang Li, "A survey of complex-valued neural networks," 2021.

[7] Sven Buchholz and Gerald Sommer, "On clifford neurons and clifford multi-layer perceptrons," *Neural Networks*, vol. 21, no. 7, pp. 925–935, 2008.

[8] E. Study, *Geometrie der Dynamen*, Teubner, 1901.

[9] P. Baseilhac, S. Galice, P. Grangé, and M. Rausch de Traubenberg, "Extended complex trigonometry in relation to integrable 2d-quantum field theories and duality," *Physics Letters B*, vol. 478, no. 1-3, pp. 365–372, apr 2000.

[10] F. M. Dimentberg, "The screw calculus and its applications in mechanics," 1968, WP-AFB, Ohio.

[11] A. Ceylan Çöken and Ali Görgülü, "On the dual darboux rotation axis of the dual space curve," *Demonstratio Mathematica*, vol. 35, no. 2, pp. 385–390, 2002.

[12] Atilim Gunes Baydin et al., "Automatic differentiation in machine learning: a survey," 2015.

[13] Ravi Kiran and Kapil Khandelwal, "Automatic implementation of finite strain anisotropic hyperelastic models using hyper-dual numbers," *Com. Mech.*, vol. 55, 2014.

[14] Yuto Okawa and Tohru Nitta, "Learning properties of feedforward neural networks using dual numbers," in *2021 APSIPA ASC*, 2021, pp. 187–192.

[15] Dmitry Kozlov, Stanislav Pavlov, Alexander Zuev, Mikhail Bakulin, Mariya Krylova, and Igor Kharchikov, "Dual-valued neural networks," in *2022 18th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*, 2022, pp. 1–8.

[16] Sergey Ioffe and Christian Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Proceedings of the 32nd ICML*, 2015, p. 448–456.

[17] Chiheb Trabelsi et al., "Deep complex networks," 2018.

[18] Judith Brown, "Calculation of a constant Q spectral transform," *JASA*, vol. 89, pp. 425, 01 1991.

[19] G. Levkine, "Prewitt, sobel and scharr gradient 5x5 convolution matrices," 2012.

[20] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *2016 IEEE CVPR*, 2016, pp. 770–778.

[21] J. Thickstun, Z. Harchaoui, and S. M. Kakade, "Learning features of music from scratch," *ICLR 2017*, pp. 1—14, 2017.