

ТЕХНИЧЕСКИЕ НАУКИ

УДК 004.8

DOI 10.34822/1999-7604-2020-2-20-31

ВЕЙВЛЕТ-ПОДОБНАЯ АРХИТЕКТУРА КОМПЛЕКСНОЗНАЧНОЙ СВЕРТОЧНОЙ НЕЙРОННОЙ СЕТИ ДЛЯ СИНТЕЗА КОМПЛЕКСНЫХ СИГНАЛОВ

Д. А. Караваев

*Санкт-Петербургский государственный университет телекоммуникаций
им. проф. М. А. Бонч-Бруевича, Санкт-Петербург, Россия
E-mail: d.a.karavaev@yandex.ru*

В работе предложена архитектура комплекснозначной сверточной нейронной сети, разработанная на основе структуры дискретного вейвлет-преобразования. Данная архитектура позволяет производить многоуровневую декомпозицию комплексного сигнала, формируя набор признаков, который можно применять для задач синтеза и классификации сигналов. Приведены результаты решения задачи предсказания значений хаотического комплексного сигнала нейронной сетью, основанные на предлагаемой архитектуре. Полученные результаты сравниваются с результатами решения данной задачи при помощи вещественнозначных нейросетевых моделей, основанными на альтернативных современных подходах. Также проведен анализ представленной комплекснозначной сверточной сети в частотной области, осуществленный за счет сравнительно небольшого числа адаптивных параметров.

Ключевые слова: цифровая обработка сигналов, машинное обучение, комплекснозначные нейронные сети, вейвлет-преобразование, прогнозирование временных рядов.

WAVELET-LIKE ARCHITECTURE OF COMPLEX-VALUED CONVOLUTIONAL NEURAL NETWORK FOR COMPLEX SIGNAL SYNTHESIS

D. A. Karavaev

*Bonch-Bruevich Saint-Petersburg State University of Telecommunications,
Saint Petersburg, Russia
E-mail: d.a.karavaev@yandex.ru*

The paper proposes the architecture of a complex-valued convolutional neural network, built upon a structure of the discrete wavelet transform. This architecture allows performing multiscale decomposition of a complex signal, thus forming a set of features that can be used for the tasks of signal synthesis and classification. The article presents the results of solving the prediction problem for the values of a chaotic complex signal by a neural network based on the proposed architecture. The obtained results are compared with the results of solving this problem using real-valued neural networks based on alternative modern approaches. The analysis of the presented complex-valued convolutional network in the frequency domain is also carried out, which was achieved due to a relatively small number of adaptive parameters.

Keywords: digital signal processing, machine learning, complex-valued neural networks, wavelet transform, time series forecasting.

Введение

Мотивация. На сегодняшний день нейросетевые адаптивные алгоритмы находят свое успешное применение в широком спектре задач обработки сигналов различной природы [1]. Подавляющее большинство нейронных сетей, решающих подобные задачи, являются веще-

ественнозначными, то есть в качестве параметров модели используются вещественные числа, и анализу подлежат вещественные сигналы. Однако существует множество потенциальных приложений, в которых сигналы имеют комплексную природу. Например, задачи, встречающиеся в радиотехнике и связи [2]. Использование в таких случаях вещественнозначной нейронной сети ведет к потере явной информации о фазе сигнала, так как вещественные и мнимые части отсчетов сигнала трактуются подобной моделью как два независимых канала. Таким образом, возникает необходимость в применении комплекснозначных нейронных сетей (КНС), в которых параметры являются комплексными числами, и все операции с ними проводятся согласно соответствующим определениям для комплексных чисел.

Краткий обзор современных достижений. Вопрос разработки КНС имеет длительную историю [3], однако в связи с последними успехами и открытиями в области глубокого обучения данный вопрос актуализировался в новом свете. Так, в статье [4] представлены комплексные аналоги вещественным базовым блокам для построения современных нейронных сетей: различные функции активации, пакетная нормализация [5], процедуры инициализации параметров, et cetera. Особенно стоит выделить работу [6], в которой представлен теоретический анализ специального типа сверточной КНС и отмечается сходство описываемой модели с процедурой дискретного вейвлет-преобразования (ДВП). Некоторые принципы, описанные в отмеченной статье, были применены для разработки архитектуры комплекснозначной сверточной нейронной сети (далее – КСНС), представленной ниже.

Условные обозначения. В тексте скалярные величины представлены курсивными строчными символами, например a . Сигналы и их значения обозначаются аналогично, совместно с временным индексом, заключенным в квадратные скобки: $a[n]$. Вектора и матрицы представлены строчными и заглавными буквами жирным шрифтом без курсива, например $\mathbf{a}$ и $\mathbf{A}$ соответственно. Множества комплексных и вещественных чисел обозначаются соответствующими символами $\mathbb{C}$ и $\mathbb{R}$. Для числа $z \in \mathbb{C}$ вещественная часть обозначается как $\operatorname{Re}(z)$, а мнимая – $\operatorname{Im}(z)$. Мнимая единица – $j = \sqrt{-1}$. Операторы, применяемые к сигналам и векторам, представлены заглавными каллиграфическими буквами с аргументом, помещенным в фигурные скобки: $\mathcal{A}\{a[n]\}$ или $\mathcal{A}\{\mathbf{a}\}$.

Постановка задачи. Цель данной работы заключается в построении архитектуры КСНС для задачи синтеза комплексных сигналов, которая состоит в аппроксимации некоторого оператора $\mathcal{G}$, сопоставляющего сигналу $s[n]$ сигнал $g[n]$:

$$\mathcal{G}\{s[n]\} = g[n]. \quad (1)$$

На практике предполагается, что в качестве входа в (1) подается фрагмент $s[n]$ конечной длительности N – вектор $s[n] = (s[n - N + 1], s[n - N + 2], \dots, s[n])$. Таким образом (1) переходит в

$$\mathcal{G}\{\mathbf{s}[n]\} = \mathbf{g}[n], \quad (2)$$

где $\mathbf{g}[n]$ – вектор конечной размерности, в общем случае не равной N . Результат применения некоторой модели $\mathcal{G}$, аппроксимирующей $\mathcal{G}$, будет именоваться $\mathbf{g}[n]$.

В качестве тестовой задачи для оценки применимости предлагаемой нейросетевой архитектуры к проблемам, формулируемым в виде (2), была выбрана задача одношагового предсказания значений $s[n]$, для которой

$$\mathcal{G}\{\mathbf{s}[n]\} = \mathbf{g}[n] \equiv \mathbf{s}[n+1]. \quad (3)$$

При использовании $\mathcal{G}$, результатом в (3) будет $\tilde{s}[n+1] = (\tilde{s}[n-N+2], \tilde{s}[n-N+3], \dots, \tilde{s}[n+1])$ – вектор, состоящий из оценок $\tilde{s}[n]$ соответствующих значений $s[n]$. Следовательно, имея решение задачи (3), можно осуществить *рекурсивное (многошаговое) предсказание* $\tilde{s}[n+K]$ путем ввода в аппроксимирующую модель $\mathcal{G}$ вектора $\tilde{s}[n+K-1]$.

Инструменты для моделирования. Компьютерные эксперименты были осуществлены с использованием связки программных библиотек Theano [7] и Keras [8] для языка программирования Python. В ходе экспериментов использовалась программная библиотека, разработанная авторами статьи [4], которая предоставляет базовый функционал для работы с комплекснозначными сетями в рамках функционала Keras и Theano. Все графики, представленные ниже, были построены при помощи библиотеки Matplotlib [9] для языка Python. Исходный код на языке Python и параметры нейросетевых моделей доступны по ссылке: URL: <https://github.com/dkaravaev/wavelet-ccnn>.

Базовый блок обработки

Основным блоком обработки предлагаемой архитектуры КСНС является блок, реализующий нелинейную фильтрацию сигнала $s[n]$, включающий в себя три этапа (рис. 1).

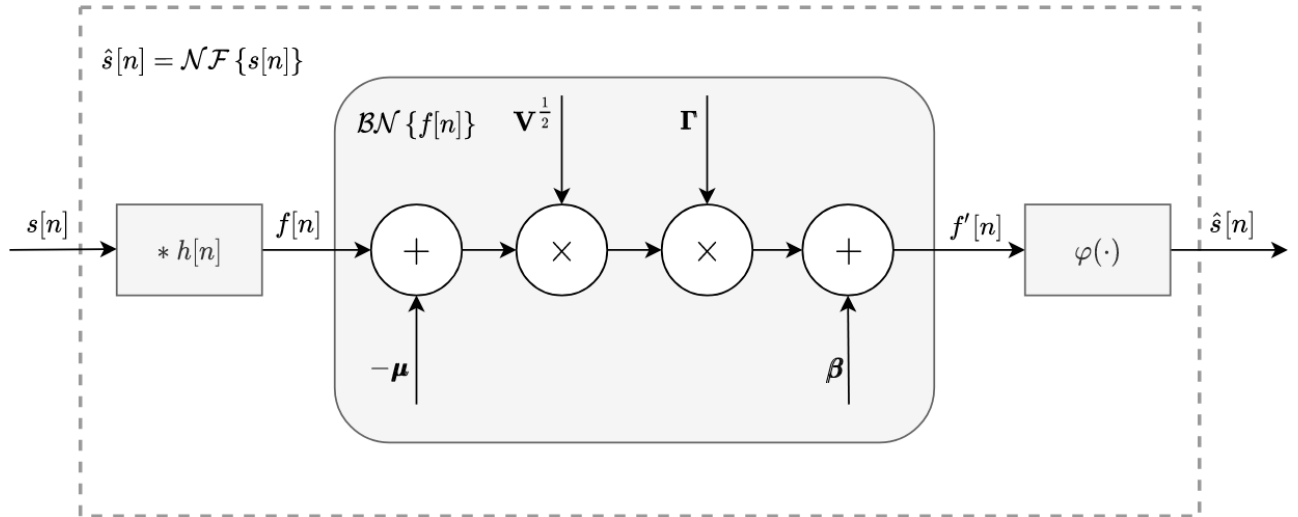


Рис. 1. Схема блока обработки

Примечание: составлено автором.

Частотная фильтрация. Первым этапом является частотная фильтрация входного сигнала $s[n]$, которая осуществляется на основе его свертки с комплексной конечной импульсной характеристикой (ИХ) $h[n]$ длины T_h , отсчеты которой подлежат обучению:

$$f[n] = s[n] * h[n] = \sum_{k=0}^{T_h-1} \bar{h}[k] s[n-k], \quad (4)$$

где под $\bar{h}[k]$ понимается число, комплексно сопряженное к $h[k]$.

В некоторых случаях перед блоком обработки может использоваться процедура децимации (понижении частоты дискретизации) с коэффициентом p , которая осуществляется неявно за счет расширяющейся свертки (dilated convolution) со степенью, равной p , из-за чего формула (4) в общем случае принимает вид:

$$f_p[n] = s_p[n] * h[n] = \sum_{k=0}^{T_h-1} \bar{h}[k] s[n-pk]. \quad (5)$$

Использование расширяющейся свертки (5) позволяет проводить многомасштабный анализ входного сигнала за счет экспоненциального расширения области применения фильтра, или поля восприятия (receptive field) [10]. При оперировании фрагментом $s[n]$ конечной длительности N ($s[n]$) для того, чтобы результат (5) также имел длительность N , вектор фрагмента дополняется в начале $p(T_h - 1)$ нулями. Далее результат фильтрации из уравнения (5) будет обозначаться как $f[n]$ для упрощения нотации. В случаях, когда будет применяться децимация, значение $p \neq 1$ будет указано явно.

Пакетная нормализация. После частотной фильтрации осуществляется этап пакетной нормализации $f'[n] = \text{BN}\{f[n]\}$ (batch normalization) [7], суть которого заключается в центрировании и перенормировке отсчетов сигнала $f[n]$. Данная процедура происходит согласно алгоритму, предложенному в статье [6], который вместо комплексных отсчетов $f[n]$ и $f'[n]$ оперирует соответствующими вещественными двумерными векторам $\mathbf{x} = (\text{Re}(f[n]), \text{Im}(f[n]))$ и $\mathbf{x}' = (\text{Re}(f'[n]), \text{Im}(f'[n]))$:

$$\mathbf{y} = \mathbf{V}^{-\frac{1}{2}}(\mathbf{x} - \boldsymbol{\mu}), \quad (6)$$

$$\mathbf{x}' = \boldsymbol{\gamma}\mathbf{y} + \boldsymbol{\beta}, \quad (7)$$

где $\mathbf{V}, \boldsymbol{\mu}$ – выборочные (пакетные) оценки матрицы ковариации и математического ожидания вектора $\mathbf{x}$, а $\boldsymbol{\gamma}, \boldsymbol{\beta}$ – квадратный корень матрицы ковариации и математическое ожидание вектора $\mathbf{x}$, которые будут найдены в процессе обучения. Подробное описание и обоснование данного алгоритма можно найти в оригинальной статье [6]. Включение данной процедуры в цепочку обработки приводит к значительному ускорению сходимости модели при поиске значений адаптивных параметров, однако на сегодняшний день строгого обоснования эффективности данного метода не представлено [11].

Нелинейная функция. Заключительным этапом в базовом блоке обработки является нелинейная функция активации $\varphi(\cdot)$, которая применяется к каждому отсчету сигнала $f'[n]$. В данной работе используется тривиальное обобщение функции активации ELU(x), $x \in \mathbb{R}$ (Exponential linear unit) [12] для случая комплексных чисел – ELU(z), $z \in \mathbb{C}$:

$$\text{ELU}(x) = \begin{cases} x, & x > 0 \\ \alpha(e^x - 1), & x \leq 0 \end{cases}, \quad (8)$$

$$\text{ELU}(z) = \text{ELU}(\text{Re}(z)) + j\text{ELU}(\text{Im}(z)), \quad (9)$$

где параметр $\alpha \in \mathbb{R}$ задается a priori (далее $\alpha = 1$). Применение такой функции активации продиктовано результатами проведенных компьютерных экспериментов, в которых ее использование приводило к улучшению сходимости КСНС по сравнению со случаями использования иных функций, в том числе учитывающих специфику комплексных чисел – modReLU(z) [13].

Итоговая формула. Применение к входному сигналу подобного блока нелинейной фильтрации (оператора) будет обозначаться следующим образом:

$$\hat{s}[n] = \mathcal{NF}\{s[n]\}. \quad (10)$$

Архитектура нейросетевой модели

Этап анализа. На основе предложенного блока обработки (10) можно сформировать КСНС с топологией, основанной на структуре процедуры ДВП [14]. При данной топологии блоки фильтрации соединяются в пару, формируя нелинейный банк фильтров, который также именуется *сверточным слоем* нейронной сети [15]. После чего происходит рекуррентное

преобразование выхода с 0-го блока обработки, который предварительно был децимирован с коэффициентом 2, до заданного уровня L (число слоев) (рис. 2).

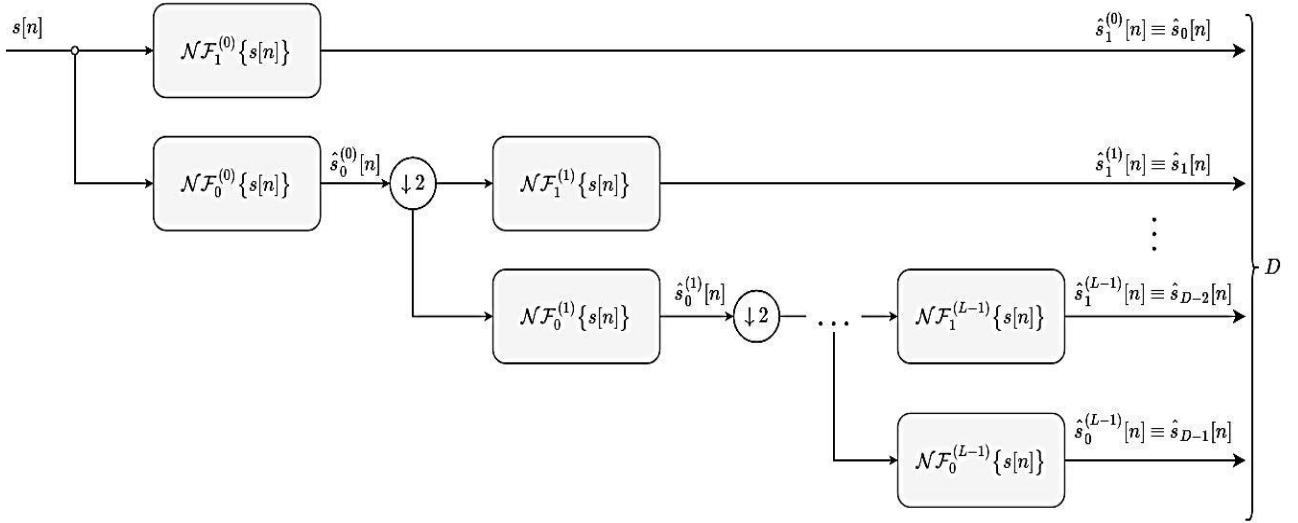


Рис. 2. Схема КСНС с вейвлет-подобной архитектурой, реализующей этап анализа

Примечание: составлено автором.

Как уже было отмечено выше, децимация происходит за счет использования расширяющейся свертки, таким образом, во всех слоях, начиная с первого, применяется свертка в виде (5) с $p = 2$. Полученная сверточная нейронная сеть осуществляет декомпозицию входного сигнала на $D = L + 1$ каналов – $\hat{s}_0[n], \hat{s}_1[n], \dots, \hat{s}_{D-1}[n]$ по принципу, схожему с ДВП, в процессе которого осуществляется рекурсивная частотная фильтрация парой фильтров с заданными коэффициентами – фильтрами нижних и высоких частот (ФНЧ и ФВЧ соответственно). Далее сети с такой организацией блоков обработки (10) будут называться сетями с вейвлет-подобной архитектурой. Авторы статьи [6] предлагают называть данное преобразование дискретным нелинейным вейвлет-преобразованием. Также стоит отметить, что подобную архитектуру можно обобщить для случая с числом фильтров более двух в одном слое.

С позиции теории цифровой обработки сигналов подобную сверточную сеть можно трактовать как нелинейный банк фильтров, реализующий этап анализа.

Для сравнения: в стандартных архитектурах сверточных нейронных сетей используется иной принцип стыковки слоев, суть которого заключается в том, что каждый фильтр в l -м слое имеет ИХ для каждого входного канала (выходного с $l-1$ слоя), и результатом применения такого фильтра является сумма сверток каждого входного канала с соответствующей ИХ. Преимущество предлагаемого метода состоит в том, что информация с этапов первичной обработки непосредственно отображается в результирующем наборе каналов, что позволяет расширить пространство признаков для дальнейшего обучения (например, полносвязных слоев) при меньшем числе фильтров в каждом слое и улучшить сходимость модели при обучении методом обратного распространения ошибки [16].

Результат применения КСНС с вейвлет-подобной архитектурой (декомпозиции) к входному сигналу $s[n]$ далее будет обозначаться как $WD_{s[n]}$.

Этап синтеза. Для решения задачи синтеза (1) на основе описанной выше декомпозиции $WD_{s[n]}$ можно воспользоваться методом, предложенным в работах [17, 18], в котором сигнал $g[h]$ получается путем взвешенного суммирования по всем полученным каналам (рис. 3):

$$g[n] = \sum_{d=0}^{D-1} v_d \hat{s}_d[n], \quad (11)$$

где $v_d \in \mathbb{C}$ для $d \in \{0, 1, \dots, D-1\}$ – набор адаптивных параметров. С точки зрения сверточных нейронных сетей операцию (11) можно представить как применение стандартного сверточного слоя с длинами ИХ для каждого частотного канала, равными 1 (1×1 convolution).

Таким образом, процедура (11) является этапом *синтеза* с точки зрения терминологии цифровой обработки сигналов. Итоговая схема КСНС для синтеза $\tilde{g}[n]$, приведена на рис. 3.

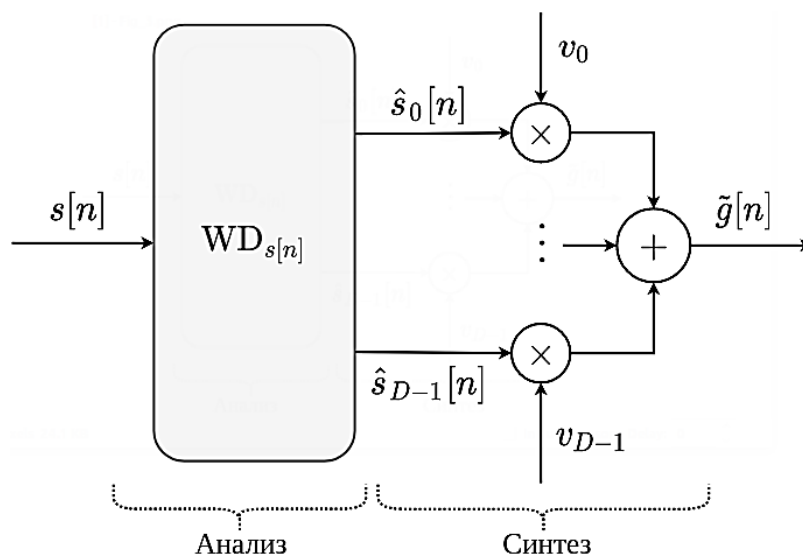


Рис. 3. Схема вейвлет-подобной КСНС для задачи синтеза сигналов

Примечание: составлено автором.

Обучение модели

Формализация задачи обучения. Первичным этапом обучения модели является формирование пар $(s[n], g[n])$, то есть пар входа и желаемого выхода модели $\tilde{g}$. Далее данные пары разбиваются на обучающую и тестовую выборки, первая из которых служит непосредственно для нахождения значений параметров модели, а вторая – для проверки достижения моделью *обобщающего* свойства [15].

Формально поиск значений адаптивных параметров нейронной сети для синтеза $\tilde{g}[n]$ (2) происходит в процессе минимизации функции среднеквадратической ошибки (ФСО) для различных фрагментов (batch) обучающей выборки:

$$L_{\text{MSE}} = \frac{1}{M} \sum_{m=0}^{M-1} \| \mathbf{g}_m - \tilde{\mathbf{g}}_m \|^2, \quad (12)$$

где M – размер фрагмента. Временные индексы у $\mathbf{g}_m$ опущены, так как эти вектора, формирующие фрагменты, выбираются из обучающей выборки случайным образом.

Во избежание эффекта переобучения к функции (12) добавляется слагаемое регуляризации типа L^2 :

$$L_{\text{re}} = \sum_{l=0}^{L-1} (\|h_0^{(l)}\|^2 + \|h_1^{(l)}\|^2) + \sum_{d=0}^{D-1} |v_d|^2, \quad (13)$$

где $h_i^{(l)}$ – ИХ i -го фильтра ($i \in \{0,1\}$) в l -м слое ($l \in \{0, \dots, L-1\}$) и v_d – коэффициенты из (11).

Таким образом, учитывая (13), итоговая форма функции ошибки на одном фрагменте обучающей выборки имеет вид:

$$L = L_{\text{MSE}} + L_{\text{re}}, \quad (14)$$

где λ_{re} – коэффициент регуляризации.

Решение. Минимизация (14) осуществляется на основе итеративной процедуры градиентного спуска, в котором обучаемые параметры модели обновляются по правилу:

$$w_{\tau+1} = w_{\tau} - \eta \frac{\partial L(w)}{\partial w}, \quad (15)$$

где w – некоторый параметр модели, τ – номер итерации обучения, и η – коэффициент скорости обучения (learning rate). Вычисление производной в формуле (15) осуществляется на основе алгоритма обратного распространения ошибки. Если $w \in \mathbb{C}$, формула (15) применяется для $\text{Re}\{w\}$ и $\text{Im}\{w\}$ соответственно. В экспериментах, описанных далее, был использован модифицированный вариант формулы обучения (15), который носит название Nadam [19].

Немаловажным пунктом в процессе обучения нейронной сети является этап инициализации адаптивных параметров [20] для процедуры (15). В данной работе, согласно статье [6], инициализация некоторого комплексного параметра w происходит в два этапа. В первую очередь $|w|$ присваивается случайное значения из распределения Рэлея с параметром $\sigma = 1/(\sqrt{n_i} + \sqrt{n_o})$ по правилу Глорота [20], где n_i и n_o – число входов и выходов в одном слое. На втором этапе $\arg\{w\}$ инициализируется случайным образом из равномерного распределения на отрезке $[-\pi, \pi]$.

Предсказание модельного сигнала

Модельный сигнал. В качестве сигнала $s[n]$ для прогнозирования (3) была выбрана хаотическая траектория отображения Икеда [21], которая задается формулой:

$$z_1[n] = A + Bz_1[n-1] \exp \left\{ \frac{jK}{|z_1[n-1]|^2 + 1} + C \right\} \quad (16)$$

где $z_1[n] \in \mathbb{C}$ – значение траектории в момент времени n и $A, B, K, C \in \mathbb{R}$ – параметры, варьируя которые можно получать различное поведение динамической системы.

Конкретная траектория $s[n] \equiv z_1[n]$ длительностью 40 000 отсчетов была получена из системы (16) со значениями параметров $A = 1$, $B = 0,9$, $K = -6$, $C = 0,4$ и начальным условием $z_1[0] = 0,5 + j$. Портрет $z_1[n]$ на комплексной плоскости представлен на рисунке 4. Сигнал $s[n]$ характерен для высокой степени нерегулярности, что видно на рисунке 5, на котором изображен модуль автоковариационной функции $C_s[q]$ (АФ) – его первых 2 000 отсчетов для положительных временных лагов $q \geq 0$. Поэтому данный сигнал является привлекательным с точки зрения его использования для изучения свойств сходимости адаптивных моделей.

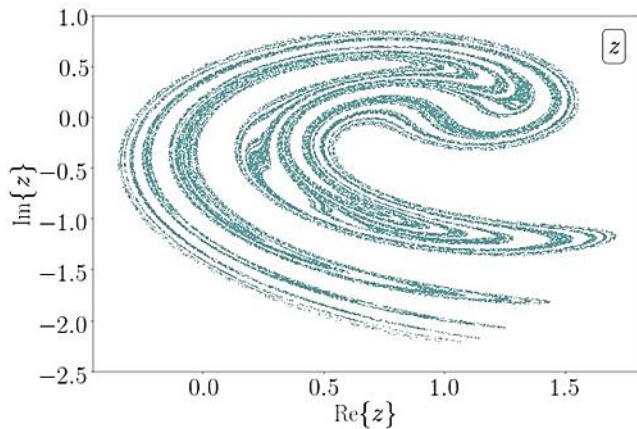


Рис. 4. Портрет траектории Икеда
Примечание: составлено автором.

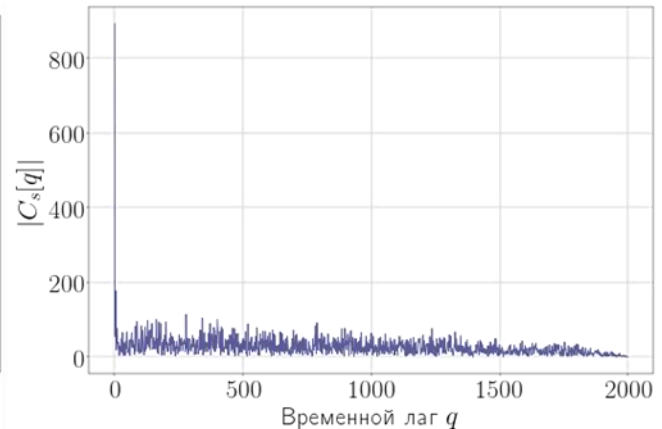


Рис. 5. Модуль АФ для траектории Икеда
Примечание: составлено автором.

Для проверки качества получаемых предсказаний будет использоваться коэффициент предсказательной силы (prediction gain) [22], измеряемый в децибелах (дБ):

$$R = 10 \log_{10} \left(\frac{\sigma_s^2}{\sigma_e^2} \right), \quad (17)$$

где σ_s^2 и σ_e^2 – оценки дисперсий сигнала $s[n]$ и сигнала ошибки $e[n] = s[n] - \tilde{s}[n]$ соответственно.

Обучающая и тестовая выборки. В формировании обучающей и тестовой выборок участвовали первые 24 000 отсчетов $s[n]$, тогда как последующие 16 000 были отложены для оценки работы модели за пределами этих выборок. Каждый фрагмент обучающей выборки содержал $M = 100$ векторов, длительность которых – $N = 1024$.

Альтернативные нейросетевые модели. Для сравнения качества решения поставленной задачи вейвлет-подобной КНС были введены две вещественнозначные нейронные сети, спроектированные на базе современных архитектурных концепций:

1) Полностью сверточная нейронная сеть (ПНС), состоящая из 8 сверточных слоев со степенью расширения свертки $p = 2$ и последнего слоя, осуществляющего суммирование в виде (11) для вычисления $\tilde{s}[n+1]$;

2) Комбинированная нейронная сеть, состоящая из двух различных концептуальных блоков. Первый блок состоит из двух сверточных слоев и формирует набор признаков для второго блока, которым является рекуррентная нейронная сеть типа LSTM [23], осуществляющая прогнозирование. Такая модель будет условно обозначаться ниже как СНС + LSTM.

В обеих сетях сверточные слои имели 32 фильтра, длина ИХ которых равна 4. После каждого сверточного слоя применялись пакетная нормализация и функция активации (8). Общее число адаптивных параметров в ПНС и СНС + LSTM составило 29 729 и 4 824 соответственно. В процессе обучения для этих моделей полагалось, что $\lambda_{ге} = 0,001$ (13) и $\eta = 0,001$ (15).

Критерием сравнения между вейвлет-подобной КНС и данными сетями будет выступать значение функции L_{MSE} (12) на тестовой выборке.

Результаты. Оптимальной из обученных КНС с вейвлет-подобной архитектурой стала модель с глубиной декомпозиции $L = 6$ и $T_h = 4$ (5), для которой коэффициенты скорости обучения и регуляризации – $\lambda_{ге} = 0,002$ (13) и $\eta = 0,001$ (15). Таким образом, модель содержит 85 комплексных или 170 вещественных адаптивных параметров. В решении данной регрессионной задачи представленная КНС значительно превосходит по качеству вещественнозначные модели ПНС и СНС + LSTM, что можно увидеть на рис. 6, на котором

изображена динамика значений L_{MSE} , полученная на тестовой выборке для каждой из нейронных сетей. При этом КСНС обладает существенно меньшим числом адаптивных параметров по сравнению с вещественнозначными альтернативами.

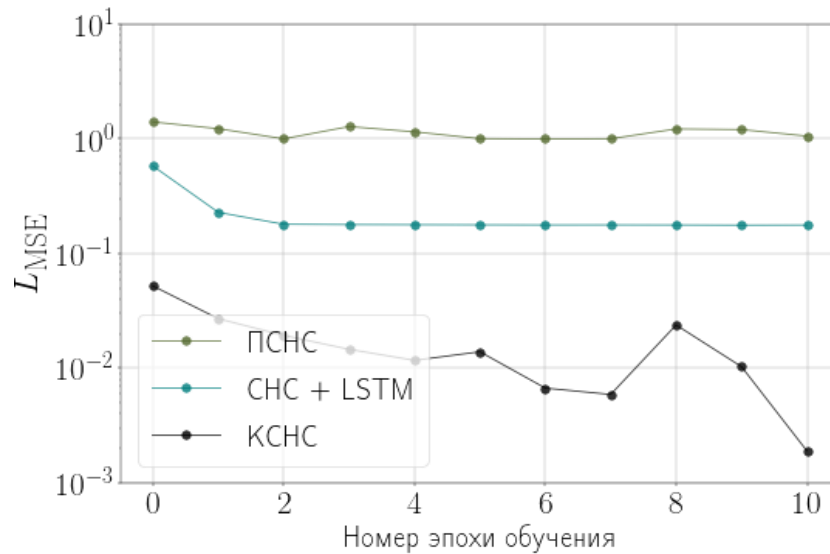


Рис. 6. Значения ФСО на тестовой выборке для КСНС, ПСНС и СНС + LSTM
Примечание: составлено автором.

Для сигнала $\tilde{s}[n]$, который был получен путем стыковки последних элементов векторов $\tilde{s}[n+1]$ на всей длительности траектории (40 000 отсчетов), коэффициент $R \approx 22,9$ [дБ], что говорит о достижении КСНС обобщающего свойства в задаче прогнозирования на один шаг вперед. Стоит отметить, что данное значение коэффициента (18) существенно выше аналогичных значений для одношаговых предсказаний сигнала (16), полученных линейными адаптивными методами в работе [22], порядок которых составил 4 [дБ] [22, с. 4].

Однако наиболее интересным является анализ многошагового рекурсивного предсказания. Результат работы КСНС в таком режиме представлен на рис. 7 с изображением портрет сигнала $\tilde{s}_r[n]$ длительностью 40 000 отсчетов на комплексной плоскости. На данном рисунке видно, что модель не смогла повторить сложные детали портрета траектории системы Икеда (16), несмотря на общее визуальное сходство $\tilde{s}_r[n]$ и $s[n]$, коэффициент $R \approx -2,6$ [дБ], что говорит о низком качестве полученного предсказания. Можно предположить, что это является следствием неадекватности оценки низкочастотных составляющих сигнала $s[n]$ нейросетевой моделью, то есть неспособности к обучению его долгосрочным зависимостям. Данный факт проявляется при сравнении оценок спектральных плотностей мощности (СПМ) сигналов $\tilde{s}_r[n]$ и $s[n]$, представленных на рис. 8. Приведенные оценки СПМ были получены на основе метода Уэлша с числом частотных ячеек, равным 1 024. Предварительно оба сигнала были центрированы для удаления пика на нулевой частоте.

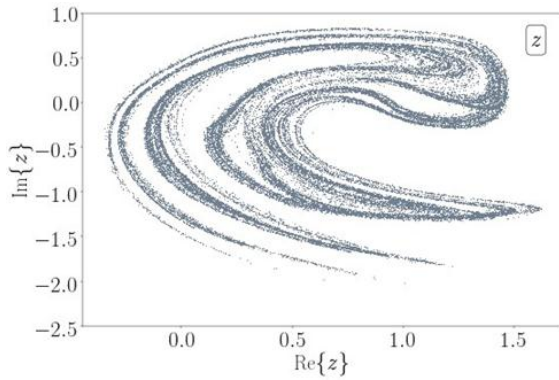


Рис. 7. Портрет рекурсивного предсказания
Примечание: составлено автором.

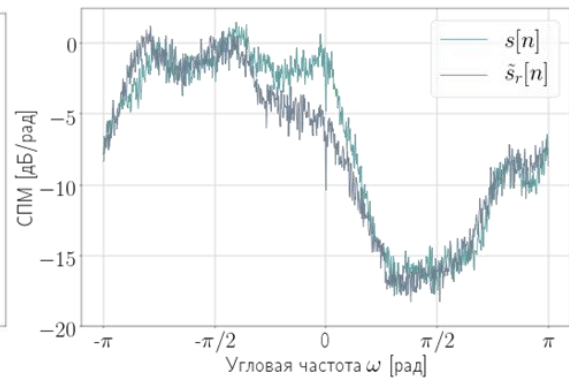


Рис. 8. Оценки СПМ
Примечание: составлено автором.

Также представляется интересным провести качественное сравнение фильтров полученной КСНС с фильтрами ДВП в разрезе частотно-избирательных свойств. Схема с амплитудно-частотными характеристиками (АЧХ) фильтров КСНС представлена на рис. 9. На данной схеме видно, что фильтры во всех слоях, кроме последнего, не обладают значительными частотно-избирательными свойствами. Только фильтры в последнем слое можно нестрого трактовать как пару ФНЧ и ФВЧ.

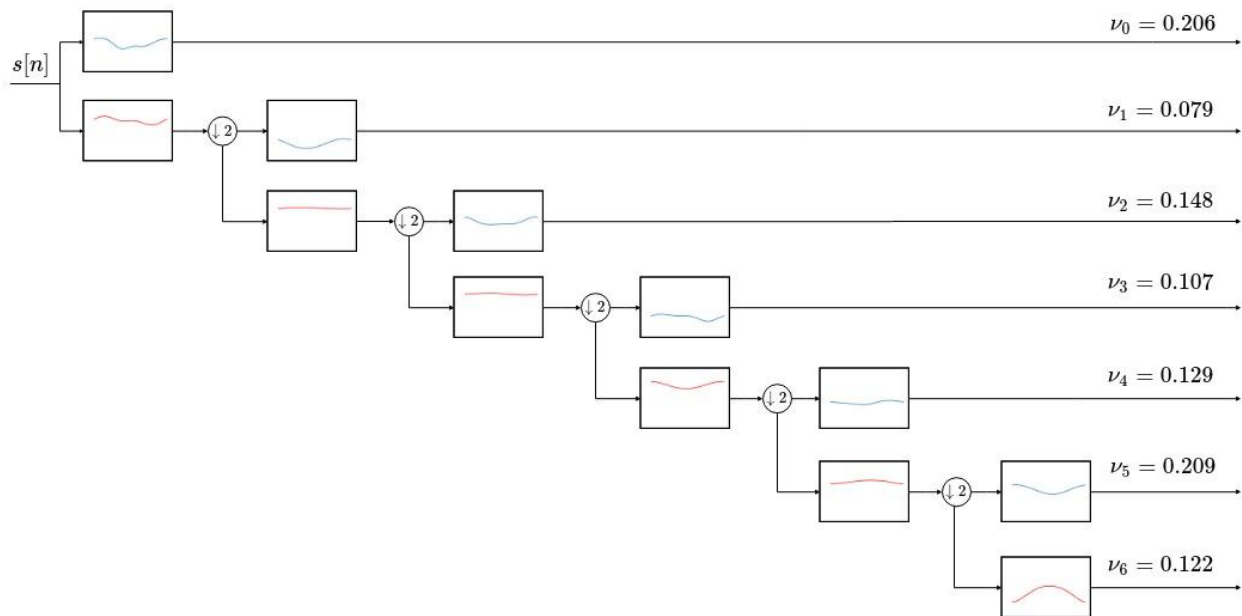


Рис. 9. АЧХ фильтров КСНС с вейвлет-подобной архитектурой для прогнозирования траектории системы Икеда; Значения АЧХ представлены в линейной шкале на отрезке [0; 0.5]
Примечание: составлено автором.

На рис. 9 у выхода каждого фильтра, результаты которого включаются в итоговый набор каналов, представлена величина:

$$\nu_d = \frac{1}{V} |v_d|, \quad v = \sum_{d=0}^{D-1} |v_d|, \quad (18)$$

где параметры ν_d из формулы (11). Значение ν_d отражает степень участия d -го канала в синтезе оценки $\tilde{s}[n+1]$. Величины ν_d оказались распределены относительно равномерно, следовательно, КСНС равноправно использует информацию с выходов результирующих каналов.

Заключение

Основным результатом данной работы является предложенная нейросетевая архитектура, осуществляющая декомпозицию сигнала, структурно схожую с дискретным вейвлет-преобразованием. Причем результат подобного разбиения возможно применять не только для задач синтеза сигналов, таких как прогнозирование, но и для их классификации.

Приведенная выше модель, основанная на вейвлет-подобной архитектуре, успешно справилась с задачей синтеза одношагового предсказания хаотического сигнала при относительно малом числе адаптивных параметров, показав при этом значительно лучший результат по сравнению со стандартными линейными и на порядок более сложными вещественно-значными нейросетевыми аналогами. Однако найденная модель оказалась неспособной решить задачу долгосрочного прогнозирования с удовлетворительным качеством.

В дальнейшем есть несколько путей усовершенствования таких моделей. Во-первых, можно задавать тип избирательности фильтра в слое, как это было предложено в работе [8], таким образом явно указывая на разбиение полосы входного сигнала. Потенциально это может помочь в случае анализа сигналов с заранее известными полосовыми характеристиками. Во-вторых, можно модифицировать этап синтеза сигнала. Например, использовать симметричную схему канальной композиции согласно структуре восстановления сигнала в дискретном вейвлет-преобразовании. Это может обогатить модель и позволит избежать избыточного добавления нулей на каждом этапе свертки для поддержания равенства длительностей входного и выходного сигнала в тех задачах, где это необходимо.

В перспективе подобные комплекснозначные сверточные нейронные сети могут служить альтернативой стандартным адаптивным методам в радиотехнических задачах, таких как компенсация искажений, вносимых каналом в системах связи [2], или подавление узкополосных помех на фоне широкополосного приема [24].

Литература

1. LeCun Y., Bengio Y., Hinton G. E. Deep Learning // Nature. 2015. Vol. 521, No. 7553. P. 436–444. DOI: <https://doi.org/10.1038/nature14539>.
2. Джиган В. И. Адаптивные фильтры и их приложения в радиотехнике и связи. Ч. 3 // Соврем. электроника. 2010. Т. 2. С. 70–77.
3. Hirose A. Complex-Valued Neural Networks. Studies in computational intelligence. Springer-Verlag, 2006. URL: <https://books.google.ru/books?id=d2INKuuzQjEC> (дата обращения: 30.04.2020).
4. Chiheb T., Bilaniuk O., Serdyuk D., et al. Deep Complex Networks // CoRR. 2017. URL: <http://arxiv.org/abs/1705.09792> (дата обращения: 30.04.2020).
5. Ioffe S., Szegedy C. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift // CoRR. 2015. URL: <http://arxiv.org/abs/1502.03167> (дата обращения: 30.04.2020).
6. Bruna J., Chintala S., LeCun Y., et al. A Theoretical Argument for Complex-Valued Convolutional Networks // CoRR. 2015. URL: <http://arxiv.org/abs/1503.03438> (дата обращения: 30.04.2020).
7. Theano Development Team. Theano: A Python framework for fast computation of mathematical expressions // arXiv e-prints. 2016. URL: <http://arxiv.org/abs/1605.02688> (дата обращения: 30.04.2020).
8. Keras: the Python deep learning API. URL: <https://keras.io>. (дата обращения: 30.04.2020).
9. Hunter J. D. Matplotlib: A 2D Graphics Environment // Computing in Science & Engineering. 2007. Vol. 9, No. 3. P. 90–95.
10. Yu F., Koltun V. Multi-Scale Context Aggregation by Dilated Convolutions // CoRR. 2015. URL: <http://arxiv.org/abs/1511.07122> (дата обращения: 30.04.2020).

11. Santurkar S., Tsipras D., Ilyas A., et al. How Does Batch Normalization Help Optimization? // *Advances in Neural Information Processing Systems*. 2018. Vol. 31, P. 2483–2493.
12. Clevert D. A., Unterthiner T., Hochreiter S. Fast and Accurate Deep Network Learning by Exponential Linear Units (ELUs) // *CoRR*. 2015. URL: <http://arxiv.org/abs/1511.07289> (дата обращения: 30.04.2020).
13. Arjovsky M., Shah A., Bengio Y. Unitary Evolution Recurrent Neural Networks // *CoRR*. 2015. URL: <http://arxiv.org/abs/1511.06464> (дата обращения: 30.04.2020).
14. Mallat S. *A Wavelet Tour of Signal Processing, Third Edition: The Sparse Way*. Orlando, FL, USA : Academic Press, Inc., 2008.
15. Goodfellow I., Bengio Y., Courville A. *Deep Learning*. MIT Press, 2016.
16. He K., Zhang X., Ren S. et al. Deep Residual Learning for Image Recognition // *CoRR*. 2015. URL: <http://arxiv.org/abs/1512.03385> (дата обращения: 30.04.2020).
17. Van den Oord A., Dieleman S., Zen H., et al. WaveNet: A Generative Model for Raw Audio // *CoRR*. 2016. URL: <http://arxiv.org/abs/1609.03499> (дата обращения: 30.04.2020).
18. Borovykh A., Bohte S. M., Oosterlee C. W. Conditional Time Series Forecasting with Convolutional Neural Networks // *CoRR*. 2017. URL: <http://arxiv.org/abs/1703.04691> (дата обращения: 30.04.2020).
19. Dozat T. Incorporating Nesterov Momentum into Adam // *Workshop track – ICLR*. 2016. URL: <https://openreview.net/pdf?id=OM0jvwB8jIp57ZJtNEZ> (дата обращения: 30.04.2020).
20. Glorot X., Bengio Y. Understanding the Difficulty of Training Deep Feedforward Neural Networks // *Proceedings of the International Conference on Artificial Intelligence and Statistics (AISTATS'10)* : Society for Artificial Intelligence and Statistics. 2010.
21. Ikeda K. Multiple-Valued Stationary State and its Instability of the Transmitted Light by a Ring Cavity System // *Optics Communications*. 1979. Vol. 30, No. 2. P. 257–261.
22. Mandic D. P., Still S., Douglas S. C. Duality between Widely Linear and Dual Channel Adaptive Filtering // *2009 IEEE International Conference on Acoustics, Speech and Signal Processing*. 2009. P. 1729–1732.
23. Hochreiter S., Schmidhuber J. Long Short-Term Memory // *Neural Computation*. 1997. Vol. 9, No. 8. P. 1735–1780.
24. Rusch L. A., Poor H. V. Narrowband Interference Suppression in CDMA Spread Spectrum Communications // *IEEE Transactions on Communications*. 1994. Vol. 42, No. 234. P. 1969–1979.