

Для случая  $\sigma = -2$  решение уравнения (2), описывающее одиночный кинк, принимает вид

$$u(x,t) = -k \left[ 1 + \tanh \left( \frac{kx - \omega t + \eta^0}{2} \right) \right], \quad (6)$$

где параметры решения  $k$  и  $\omega$  определяются из соотношений:  $\omega^2 = -k^2 - 1$  (дисперсионное соотношение) и  $4k^2 - 2k - 1 = 0$ . Выпишем явно значения параметра  $k$  :

$$k_{1,2} = (-1 \pm \sqrt{5})/4. \quad (7)$$

Замечание относительно параметра  $\eta^0$  остается в силе. Теперь уже отрицательному значению  $k$  соответствует bright антикинк, а положительному – dark кинк.

Непосредственной подстановкой можно убедиться, что соотношения (4) и (6) являются решениями уравнения (2).

Сравнение соотношений (5) и (7) показывает, что амплитудные значения топологически нетривиальных решений (4) и (6) одинаковы. Однако фаза меняется более быстро для случая  $\sigma = 1$  и эти решения раньше выходят на режим насыщения.

Полученные решения содержат только вклады от составляющей в виде кинка. Присутствующая в решении уравнения (1) солитонная составляющая теперь отсутствует. Таким образом, обобщенное уравнение ФКПП (2) позволяет описывать в среде формирование доменной структуры.

#### ЛИТЕРАТУРА

1. Пухов, А.А. Уравнение «реакция-диффузия» : учеб. пособие / А.А. Пухов. – М. : МФТИ, 2014. – 74 с.
2. Блинкова, Н.Г. Решение типа кинка модифицированного уравнения Фишера–Колмогорова–Петровского–Пискунова / Н.Г. Блинкова, М.А. Князев // Приборостроение–2020 : материалы 13 Междунар. науч.-техн. Конф., 18–20 нояб. 2020 г. – Минск : БНТУ, 2020. – С. 235–236.
3. Князев, М.А. Топологически нетривиальное решение одной модификации уравнения Фишера–Колмогорова–Петровского–Пискунова / М.А. Князев // Инновационные технологии обучения физико-математическим и профессионально-техническим дисциплинам : материалы XIII Междунар. науч.-практ. интернет-конф., 25–26 марта 2021 г. – Мозырь : МГПУ им. И.П. Шамякина, 2021. – С. 218–219
4. Абловиц, М. Солитоны и метод обратной задачи рассеяния / М. Абловиц, Х. Сегур. – М. : Мир, 1987. – 479 с.
5. Князев, М.А. Кинки в скалярной модели с затуханием / М.А. Князев. – Минск : Техналогія, 2003. – 115 с.

**А.А. КОЗИНСКИЙ**

УО БрГТУ, (г. Брест, Беларусь)

#### МЕТОДЫ ПАКЕТА LIBROSA ДЛЯ НЕЙРОСЕТЕВОЙ ОБРАБОТКИ ЗВУКА

Речевой интерфейс получил широкое распространение в системах, использующих голосовые помощники, синтезаторы речи, анализ и распознавание голоса, клонирование дикции, понимание и интерпретацию речи, голосовые переводчики, речевые экспертные системы на основе акустических характеристик, получивших название «форманта».

Представим опыт использования некоторых методов и пакетов для аудиоанализа музыки, речи и других звуков. Средства обработки звука предоставляются, например, в пакете параметризации аудио librosa [1] для языка Python, в библиотеке на основе Марковской модели Google Speech Recognition [2], модулях base64 [3], scipy [4] и др. Особенности перечисленных средств является возможность использования в сервисе Colaboratory [5]. Представим некоторые примеры их применения для обработки звука с использованием нейронных сетей.

Методы пакета librosa основаны на представлении звука в виде временного ряда. Пакет позволяет извлечь спектральные признаки аудиосигнала: центроид, спад, поток, ширину, скорость пересечения нуля, мел-частотные кепстральные коэффициенты (MFCC), цветность. Результаты применения методов librosa основаны на преобразовании звука во временной ряд и его представлении в виде массива numpy. Во фрагменте 1 представлена последовательность операций на языке Python в среде Colaboratory для получения некоторых спектральных признаков звукового файла в распространенных форматах (flac, wav и т. п.).

#### Фрагмент 1 – Фрагмент кода для извлечения спектральных признаков

```
!pip install librosa==0.8.0
import librosa
import matplotlib.pyplot as plt
import librosa.display
%matplotlib inline
# rb - временной ряд в виде массива numpy
# sr - частота дискретизации (по умолчанию 22050, изменим это на 44100)
rb, sr = librosa.load('...вписать путь к файлу', sr = 44100)
# Визуализируем тип и размерность данных
# для отображения спектральных признаков
print (type(rb), type(sr))
print (rb.shape, sr)
# Извлечение спектральных признаков
# Двумерные спектральные признаки
rb_mfccs = librosa.feature.mfcc(rb, sr) # Мел кепстральные коэффициенты
rb_chroma = librosa.feature.chroma_stft(rb, sr) # Частота цветности
# Одномерные спектральные признаки
# спектральный центроид
rb_spec_cent = librosa.feature.spectral_centroid(y = y, sr = sr)
# ширина полосы частот
rb_spec_bw = librosa.feature.spectral_bandwidth (y = y, sr = sr)
# Спектральный спад частоты
rb_rolloff = librosa.feature.spectral_rolloff (y = y, sr = sr)
# Пересечение нуля
rb_zcr = librosa.feature.zero_crossing_rate(y)
# Вычисление спектрограммы в масштабе mel
rb_spec = librosa.feature.melspectrogram(rb, sr)
```

Спектральные признаки могут быть визуализированы (см. фрагменты 2 и 3).

#### Фрагмент 2 – Фрагмент кода визуализации графика аудио

```
# Визуализируем график массива аудио
plt.figure(figsize=(14, 5))
librosa.display.waveplot(rb, sr)
```

### Фрагмент 3 – Фрагмент кода визуализации спектрограммы

```
# Визуализируем спектрограмму массива аудио на логарифмической шкале
plt.figure(figsize=(15, 7))
librosa.display.specshow(mfccs, sr=sr, x_axis='time', y_axis='log')
```

Все полученные признаки могут быть интегрированы в единый массив для подачи на вход нейронной сети для последующей обработки. В качестве такой обработки может быть выбрана задача мультиклассовой классификации звуков с целью распознавания жанра музыки, пения птиц и пр. Примером набора данных для идентификации пения птиц является «Cornell Birdcall Identification» на портале Kaggle.

Опыт автора показывает, что подготовка четырехсекундных фреймов указанного датасета из 88 классов на основе шести спектральных параметров в среде Colaboratory требует около семнадцати минут. Тогда как обучение в нейронной сети из семи слоев на 150 эпохах требует около пяти минут. Точность такой нейронной сети на проверочной выборке достигает около 67 процентов.

Рассмотренные методы пакета librosa для представления звука позволяют значительно повысить эффективность нейронных сетей на обучающих, тестовых и проверочных наборах данных в задачах обработки звука.

#### ЛИТЕРАТУРА

1. Librosa. Audio and music processing in Python [Электронный ресурс]. – Режим доступа: <https://librosa.org>. – Дата доступа: 12.01.2023.
2. SpeechRecognition 3.9.0 [Электронный ресурс]. – Режим доступа: <https://pypi.org/project/SpeechRecognition/>. – Дата доступа: 12.01.2023.
3. Base64 – Base16, Base32, Base64, Base85 Data Encodings [Электронный ресурс]. – Режим доступа: <https://docs.python.org/3/library/base64.html#module-base64>. – Дата доступа: 12.01.2023.
4. SciPy [Электронный ресурс]. – Режим доступа: <https://scipy.org/> – Дата доступа: 12.01.2023.
5. Добро пожаловать в Colaboratory! [Электронный ресурс]. – Режим доступа: <https://colab.research.google.com/>. – Дата доступа: 02.01.2023.

**А.А. КОЗИНСКИЙ, А.Л. МИХНЯЕВ**  
УО БрГТУ, (г. Брест, Беларусь)

### ОСНОВЫ РАСПОЗНАВАНИЯ ЗВУКА СРЕДСТВАМИ БИБЛИОТЕКИ SPEECH RECOGNITION

Для языка Python существует множество пакетов, которые могут быть использованы для распознавания небольших по длительности звуковых фрагментов. Одним из них является Speech Recognition [1]. Названный пакет отличает простота использования, поддержка в режимах онлайн и офлайн. Областью применения пакета может быть разработка проектов для «Умного дома» или «Интернет вещей». Пакет эффективен в распознавании диалогов из коротких фраз или команд управления устройствами.

Speech Recognition основан на выделении кепстральных коэффициентов, которые показывают зависимость мощности звукового сигнала от его частоты. Для восприятия звука органами слуха человека используется единица «мел» (см., например, [2]). Необходимо отметить, что субъективное восприятие человеком громкости звука, который определяется его амплитудой, нелинейно зависит от частоты аудио. Такая зависимость определена эмпирически [3].

Остановимся на программной реализации на языке Python применения мел-кепстральных коэффициентов, положенных в основу библиотеки Speech Recognition.