

К.Д. Рубцов<sup>1</sup>, Д.Э. Кронгауз<sup>1</sup>

<sup>1</sup> Красноярский институт железнодорожного транспорта – филиал Иркутского государственного университета путей сообщения (КрИЖТ ИрГУПС), г. Красноярск, Российская Федерация

## РАЗРАБОТКА НЕЧЁТКИХ АЛГОРИТМОВ ДЛЯ ПРОГНОЗИРОВАНИЯ И ПОТРЕБЛЕНИЯ ЭЛЕКТРОЭНЕРГИИ

**Аннотация.** Данная статья разбирает вопросы, связанные с применением нечёткой логики и математических методов для моделирования и прогнозирования потребления электрической энергии. Основная цель заключается в усовершенствовании процесса управления потреблением электроэнергии, что может снизить потери и повысить экономическую эффективность использования электричества. В процессе исследования обращается внимание на нечёткие алгоритмы, в связи с их потенциалом к адаптации к неопределённости и сложности систем электроснабжения. Статья описывает процесс разработки нечётких алгоритмов: от определения входных и выходных переменных до выбора подходящих функций принадлежности и определения правил принятия решений. Авторы приводят результаты экспериментов, проведённых для верификации разработанной нечёткой нейронной модели прогноза.

**Ключевые слова:** искусственный интеллект, нечёткая логика, прогнозирование, потребление, электроэнергия, обучение, нечёткая модель, нечёткие нейронные сети.

K.D. Rubtsov<sup>1</sup>, D. E. Krongauz<sup>1</sup>

<sup>1</sup> Krasnoyarsk Railway Transport Institute, branch of Irkutsk State Transport University, Krasnoyarsk, the Russian Federation

## DEVELOPMENT OF FUZZY ALGORITHMS FOR FORECASTING AND ELECTRICITY CONSUMPTION

**Abstract.** This article examines the issues related to the use of fuzzy logic and mathematical methods for modeling and forecasting electrical energy consumption. The main objective of this work is to improve the process of managing electricity consumption, which can reduce losses and increase the economic efficiency of electricity use. In the course of the research, attention is drawn to fuzzy algorithms, due to their potential to adapt to the uncertainty and complexity of power supply systems. The article describes the process of developing fuzzy algorithms: from the definition of input and output variables to the selection of suitable membership functions and the definition of decision-making rules. The authors present the results of experiments conducted to verify the developed fuzzy neural prediction model.

**Keywords:** artificial intelligence, fuzzy logic, forecasting, consumption, electricity, training, fuzzy model, fuzzy neural networks.

Прогнозирование и эффективное управление потреблением электроэнергии являются важными проблемами в энергетической отрасли. Для решения этих задач применяются различные математические модели и алгоритмы, в том числе нечёткие алгоритмы. Использование нечётких алгоритмов позволяет учитывать возможные неопределённости, что является неотъемлемой частью энергетического сектора.

В данной работе будет рассмотрено несколько различных нечётких алгоритмов, которые применяются для прогнозирования и управления потреблением электроэнергии. Эти алгоритмы основаны на нечёткой логике, которая позволяет работать с неопределёнными или нечеткими данными.

Один из таких алгоритмов - нечёткая логика Мамдани (один из подходов к нечёткому моделированию, предложенный при решении задач управления). Он основан на правилах, которые определяют связь между входными переменными и выходными переменными. Правила формулируются с помощью лингвистических переменных, таких как "высокий", "низкий", "средний" и т.д. (это позволяет учесть размытость или нечёткость информации, что

обычно присутствует в реальных задачах). При прогнозировании потребления электроэнергии этот алгоритм может использоваться для принятия решений о регулировании нагрузки в зависимости от текущих условий (температура, влажность, время года, день недели и т.д.).

Нечеткая логика Мамдани нашла широкое применение в области управления системами, такими как автоматическое управление трафиком, управление производственными процессами и др. Она позволяет эффективно моделировать сложные системы с размытой информацией и принимать продуктивные управленческие решения.

Ещё одним нечётким алгоритмом, который можно применить для прогнозирования и управления потреблением электроэнергии, является алгоритм адаптивной нечёткой кластеризации (ANFIS). Он комбинирует преимущества нечёткой логики и методов кластеризации, что позволяет создавать адаптивные модели, способные адекватно реагировать на изменения в потреблении электроэнергии.

Разработка и применение нечётких алгоритмов для прогнозирования и управления потреблением электроэнергии имеет большое значение в сфере энергетики. Они позволяют учитывать неопределенность и нечеткость, свойственные этой отрасли, и делать более точные и эффективные прогнозы потребления электроэнергии. Таким образом, использование нечеткой логики становится все более актуальным и востребованным в энергетической сфере [1].

Для прогнозирования электроэнергии используются различные методы, такие как методы временных рядов, искусственные нейронные сети и др. ( В данной работе рассмотрен метод прогнозирования электроэнергии с использованием нечётких алгоритмов.

В сфере энергетики существует множество факторов, влияющих на прогнозирование электроэнергии. Эти факторы могут быть неизвестными или изменяться во времени. В данном случае нечёткие алгоритмы могут быть полезными, так как они позволяют учитывать неопределённость в данных.

Для эффективного управления потреблением электроэнергии применяются различные методы оптимизации и моделирования. Кроме того, нечёткие алгоритмы могут использоваться в различных областях. Например, можно применять нечёткие контроллеры для управления системами отопления, вентиляции и кондиционирования воздуха.

Для решения данной задачи предлагается метод прогнозирования, основанный на синтезе нечёткой нейронной сети, который бы позволил прогнозировать долгосрочное потребление электроэнергии. Этот метод разработан автором статьи и включает в себя следующие алгоритмы:

А) Алгоритм формирования нечётких переменных на основе временного ряда. Он позволяет определить нечеткую зависимость между значениями потребления электроэнергии и временными параметрами.

Б) Алгоритм синтеза базы знаний. Он позволяет сформировать базу знаний, которая содержит нечеткие правила, описывающие взаимосвязи между входными переменными и выходным значением.

В) Алгоритм синтеза нечёткой нейронной сети на основе нечёткой базы знаний. Он использует нечеткую базу знаний для создания структуры нечеткой нейронной сети, которая будет выполнять прогнозирование потребления электроэнергии.

Г) Алгоритм обучения нечёткой нейронной сети. Он позволяет настроить параметры нечеткой нейронной сети на основе имеющихся данных, чтобы достичь наилучшей производительности прогнозирования.

Д) Формирование модели. По завершении обучения нечеткой нейронной сети, создается модель, которая может использоваться для прогнозирования долгосрочного потребления электроэнергии.

Е) Применение нечётких нейронных сетей для автоматического управления потреблением электроэнергии. Полученная модель может быть использована для принятия решений об оптимальном использовании электроэнергии и регулировании систем отопления, вентиляции и кондиционирования воздуха.

Таким образом, предложенный метод прогнозирования с использованием нечёткой нейронной сети может быть эффективным инструментом для управления потреблением электроэнергии.

Для разработки нечётких алгоритмов для прогнозирования и потребления электрической энергии будет использоваться язык программирования Python. Данный язык программирования является одним из самых популярных языков программирования для разработки нейронных сетей, т.к.

А) Имеет простой и понятный синтаксис;

Б) Богатая экосистема: Python имеет множество библиотек и фреймворков, которые предлагают мощные инструменты для разработки нейронных сетей;

В) Данный язык программирования можно использовать для эффективной работы с большими объёмами данных и обучения нейронных сетей на графических процессорах (GPU), т.к. Python имеет высокую производительность;

Г) Python отлично интегрируется с другими популярными технологиями, такими как базы данных, облачные сервисы, веб-фреймворки и другие. Это позволяет легко интегрировать нейронные сети с другими частями системы и использовать Python в различных промышленных и исследовательских задачах.

Для правильной работы программы нечётких алгоритмов для прогнозирования и потребления электрической энергии необходимо импортировать следующие библиотеки:

1. *Numpy* – библиотека, предоставляющая мощные средства для работы с массивами и матрицами. Она обеспечивает высокую производительность и эффективность при выполнении научных и численных операций. Numpy является одной из основных библиотек для научных вычислений в языке программирования Python и используется во многих областях, таких как машинное обучение, обработка данных, физика, биология и многие другие.

2. *Skfuzzy* – пакет для языка программирования Python, который представляет инструменты для работы с нечёткой логикой и нечёткими системами. Она предоставляет функции и классы для создания, настройки и использования моделей нечёткой логики. С помощью данного пакета можно определять нечёткие переменные, создавать нечёткие наборы и применять операции нечёткой логики, такие как импликация, агрегация и дефаззификация.

3. *Datetime* – библиотека, предоставляющая классы для работы с датами и временем. Она позволяет создавать объекты, представляющие дату, время, дату и время, а также выполнять различные операции с ними, такие как арифметические операции, форматирование и преобразование. Данная библиотека позволяет эффективно работать с датой и временем в Python и часто используется для работы с расписаниями, логированием и другими приложениями, где требуется оперировать датами и временем.

Начало разработанной программы представлено на рисунке 1.

```
File Edit Selection Find View Goto Tools Project Preferences Help
Нечёткая логика.py x
1 # РАЗРАБОТКА НЕЧЁТКИХ АЛГОРИТМОВ ДЛЯ ПРОГНОЗИРОВАНИЯ И ПОТРЕБЛЕНИЯ ЭЛЕКТРОЭНЕРГИИ
2
3 # Импорт необходимых библиотек для работы программы
4 import numpy as np
5 import skfuzzy as fuzz
6 from skfuzzy import control as ctrl
7 from datetime import datetime, timedelta
8 from datetime import date
9 from time import sleep
```

Рис. 1. Начало программы «Разработка нечётких алгоритмов для прогнозирования и потребления электроэнергии»

### 1.1 Формирование нечётких переменных на основе временного ряда

Для решения задачи с использованием методов нечёткой логики необходимо сформировать базу нечётких правил (совокупность правил, основанных на нечёткой логике, которая позволяет описывать и обрабатывать нечёткую, неоднозначную и неопределённую информацию). В этой базе содержатся правила, состоящие из условий и соотношений, которые определяют связи между нечёткими наборами данных и выводимым результатом. Это позволяет использовать нечёткие правила для принятия решений в условиях неопределённости и

нечёткости данных. В данной научной работе разработан специальный подход для формирования этих нечётных переменных[3].

Представим, что у нас имеется временной ряд  $\{x_i\}$ , где  $i = 0, \dots, N_x$ . Необходимо получить набор нечётких множеств  $\{A_j\}$  и функции принадлежности  $\mu_{A_j}(x_i)$ . В качестве функций принадлежности выбраны функции Гаусса:

$$\mu_{A_j}(x_i) = \frac{1}{1 + \left(\frac{x_i - b_j}{\sigma_j}\right)^2}, \quad (1)$$

где  $x_i$  – элемент временного ряда,  $b_j$  – центр класса нечёткого множества  $A_j$ ,  $\sigma_j$  – коэффициент пологости функции принадлежности.

Находим центры классов нечётких множеств по формуле:

$$b_j = x_{\min} + (2 \cdot j - 1) \cdot \omega, \quad (2)$$

где  $x_{\min}$  – минимальное значение временного ряда  $\{x_i\}$ , а  $\omega$  – значение шага, получаемого согласно:

$$\omega = \frac{x_{\max} - x_{\min}}{N_A}, \quad (3)$$

где  $x_{\max}$  – максимальное значение временного ряда  $\{x_i\}$ ,  $N_A$  – заданное число нечётких множеств.

## 1.2 Процедура создания базы правил.

Главной идеей этого метода является исключение экспертов из процесса формирования нечётких правил. Это достигается путём использования следующего алгоритма [4]:

*Шаг № 1.* Чтобы определить количество регрессоров в модели прогнозирования, задаётся значение глубины погружения  $\tau$ .

*Шаг № 2.* Устанавливаем текущее значение времени  $t = \tau$ .

*Шаг № 3.* Формируем левую часть нечёткого правила для текущего значения времени  $t_i$  следующим образом. Для каждого регрессора  $x_{t-p}$ , где  $p = 1, \dots, \tau$  определяется такое нечёткое множество, для которого значения функции принадлежности будет максимальным, т.е.

$$t_i = y(x_{t-p}) = \max \mu_{A_j}(x_i) \rightarrow A_n^i \quad (3)$$

*Шаг № 4.* Правая часть формируется аналогично третьему шагу, только в качестве переменной выступает текущее значение временного ряда, т.е.  $x_i$ .

*Шаг № 5.* Сохраняем правило в базе нечётких правил.

*Шаг № 6.* Если  $t = N_x$ , то завершается выполнение алгоритма, в противном случае увеличивается  $t = t+1$  и переход идёт на третий шаг.

*Шаг № 7.* Формируется из полученных термов правила для базы знаний. В контексте базы нечётких правил, термы представляют собой отдельные значения или выражения, которые могут быть нечёткими или иметь нечёткую принадлежность. Например, в базе нечётких правил для прогнозирования электрической энергии могут быть использованы: население, сезонность, температура, влажность, давление, скорость ветра, предыдущие данные потребления и т.д.

## 1.3 Процедура синтеза нечёткой нейронной сети на основе нечёткой базы правил.

Результат, полученный от предыдущего алгоритма, представляет собой базу знаний, которая содержит набор правил  $\{r_i\}$ , где  $i = 1, \dots, n_r$ , где  $n_r$  – число правил. Каждое  $i$ -е правило состоит из  $n_l^i$  термов в левой части и  $n_r^i$  в правой части. На основе этой системы правил выполняется алгоритм, который создаёт структуру нечёткой нейронной сети, состоящей из

5-ти слоёв. Обозначаем через  $x_j^{(i)}$   $j$ -ю компоненту  $i$ -го входного вектора,  $i = 1, \dots, s$ , где  $s$  – число элементов обучающей (тестовой) выборки. Выходные сигналы обозначаем как  $y_j^{(l)(i)}$ , где  $i$  – номер слоя нейронной сети,  $j$  – номер нейронной сети в слое. Для простоты обозначения опускается индекс  $(i)$ .

1. *Создание входного слоя.* Число нейронов в первом слое соответствует числу авторегрессионных переменных или числу термов в левой части нечётких правил. Значения выходов нейрона вычисляются согласно формуле:

$$y_j^{(1)} = x_j. \quad (4)$$

2. *Создание слоя фаззификации.* Количество нейронов-фаззификаторов в этом слое определяется умножением авторегрессионных переменных на число нечётких множеств, заданных для этих переменных. Например, если учитывается потребление в 2 предыдущих момента времени и определили 3 нечётких множества: «низкое», «среднее» и «высокое» потребление, то число нейронов во входном слое будет равно 6. Нейрон-фаззификатор соединяется с соответствующим нейроном входного слоя в соответствии с системой нечётких правил и выполняет нужное преобразование:

$$y_{j,k}^{(2)} = \mu_{A_k} \left( y_j^{(1)}; b_k, \sigma_k \right), \quad (5)$$

где  $k$  – номер нечёткого множества.

3. *Создание слоя минимизации.* Слой фаззификации включает в себя набор  $n_r$  нейронов. Каждый нейрон из этого слоя соответствует одному из правил нечёткой системы. Для связи с предыдущим слоем, каждый нейрон минимизации применяет структуру термов, определённых для соответствующего правила. Операция минимизации (операция произведения Мамдани) реализуется в каждом нейроне минимизатора, где происходит умножение входных сигналов [4].

$$y_j^{(3)} = \prod_{i=1}^{q_j} \left( y_i^{(2)} \right), \quad (6)$$

где  $q_j$  – число термов в левой части  $j$ -го правила.

4. *Создание слоя максимизации.* Слой фаззификации состоит из двух различных типов нейронов ( $2n$  нейронов, где  $n$  – число выходов). Каждый нейрон в слое фаззификации связан с соответствующими нейронами в предыдущем слое согласно структуре правил. У первого нейрона из пары  $y_{j,(1)}^{(4)}$  присутствуют значения весов  $\omega$ , которые соответствуют центрам классов нечётких множеств для термов в правых частях нечётких правил. У второго нейрона из пары  $y_{j,(2)}^{(4)}$  значения весов равны 1.

$$y_{j,(1)}^{(4)} = \sum_{i=1}^n \left( \omega_i y_i^{(3)} \right), \quad (7)$$

$$y_{j,(2)}^{(4)} = \sum_{i=1}^n \left( y_i^{(3)} \right). \quad (8)$$

5. *Создание выходного слоя.* Слой дефаззификации, также известный как выходной слой, имеет количество нейронов, равное числу выходов  $n$ . Каждый выходной нейрон связан с соответствующими нейронами предыдущего слоя и выполняет механизм дефаззификации относительно среднего центра [4]. Вычисление выхода производится следующим образом:

$$y_j^{(5)} = \frac{y_{j,1}^{(4)}}{y_{j,2}^{(4)}}. \quad (9)$$

## 1.4 Процедура обучения нечёткой нейронной сети.

Рассматривается процесс обучения нечёткой нейронной сети. Процедура обучения для настройки правой части нечётких правил выглядит следующим образом:

*Шаг № 1.* С использованием формул (4)-(9) в нечёткой нейронной сети определяется предсказание значения выходной переменной  $\hat{y}^m$  для каждого из  $m$  примеров обучающей выборки.

1.1 Задаётся средняя допустимая ошибка за цикл обучения  $\mathcal{E}_{don}$ , а также устанавливается скорость обучения  $\eta$ . Данный параметр будет определять, насколько быстрыми или медленными будут обновления весов модели в процессе обучения. Скорость обучения контролирует, насколько каждая итерация обучения будет корректировать веса в направлении уменьшения ошибки модели нечёткой нейронной сети. Низкая скорость обучения делает обновления более медленными и может потребовать большего числа итераций для достижения сходимости или оптимальных результатов. Высокая скорость обучения делает обновления более быстрыми, но может привести к неправильному шагу или расходимости. Подбор оптимального скорости обучения – это важный шаг в процессе обучения модели нейронной сети.

1.2 Вычисляется новое значение весов связей между 3-им и 4-ым слоем согласно следующим формулам:

$$\omega_j^m(q+1) = \omega_j^m(q) + \Delta\omega_j^m, \quad (10)$$

$$\Delta\omega_j^m = -\eta \cdot y_j^i \cdot e^m, \quad (11)$$

где  $q$  – номер цикла обучения. Один цикл обучения включает перебор всех примеров из обучающей выборки.

1.3 Вычисляется погрешность на выходе модуля.

$$e^m = \hat{y}^m - y_m, \quad (12)$$

где  $y_m$  – реальное значение.

1.4 Определяется средняя ошибка за цикл обучения.

$$me = \frac{1}{M} \sum_{m=1}^M e^m. \quad (13)$$

Если значение средней ошибки за цикл обучения превышает заданную среднюю допустимую ошибку, то происходит возврат на шаг (1.2). Цикл обучения может продолжаться до достижения требуемого качества модели или после выполнения определённого числа эпох (полных проходов по тренировочным данным). После цикла обучения модель может быть использована для предсказания на новых данных.

1.5 В противном случае процесс обучения останавливается, а нейронная сеть считается обученной.

*Шаг № 2.* Корректировка формы функций принадлежности нейронов второго слоя гауссовского типа, путём настройки центра функций принадлежности согласно следующей формуле:

$$\Delta C_i^m = -\eta \cdot e^m \cdot (\hat{y}^m - y_m) \cdot \frac{2 \cdot (x_i^m - \Delta C_i^m)}{(\sigma_i^m)^2} \quad (14)$$

Представление зависимости (10), (11) и (14) определяется способ модификации весов и параметров в предлагаемую модель на основе алгоритма обратного распространения ошибки. Данный алгоритм позволяет оптимизировать веса модели, минимизируя ошибку между предсказаниями модели и ожидаемыми результатами.

## 1.5 Формирование модели

Процесс создания модели представляет собой следующие этапы:

Предположим, что у нас есть множество  $y_{mz}$ , где  $Z = 1, \dots, z$ , где  $z$  – количество нечётких моделей. Наша задача - найти наилучшую модель с помощью следующего алгоритма:

*Шаг № 1.* Устанавливается количество нечётких моделей  $z$ .

*Шаг № 2.* Для каждой модели определяется количество нечётких множеств FuzzySets и коэффициент пологости функции принадлежности  $\sigma$ , который является постоянным.

*Шаг № 3.* Определяется множество временных отсчётов (глубина погружения)  $T\{t_1, t_2, \dots, t_z\}$ .

*Шаг № 4.* Устанавливается интервал времени между точками глубины погружения  $\lambda$ .

*Шаг № 5.* Задаются значения входных нейронов из исходного временного ряда (4). Число входных нейронов равно числу множества временных отсчётов  $T = t_z$ .

*Шаг № 6.* Устанавливается горизонт прогнозирования  $T_F$ .

*Шаг № 7.* Определяется временной ряд фактических значений.

*Шаг № 8.* Формируется нечёткая модель  $y_{mz}$ .

Результатом нечёткой модели  $y_{mz}$  является среднее возрастание выходов нечётких моделей. Данное возрастание относится к изменению величины выхода нечёткой модели во времени или с возрастом. Это показатель, который может использоваться для анализа и оценки эффективности и стабильности работы нечёткой модели. Среднее возрастание выходов нечетких моделей может быть положительным или отрицательным. Положительное среднее возрастание выходов означает, что с возрастом или во времени выход нечеткой модели увеличивается, что может указывать на улучшение производительности модели или увеличение веса признаков в процессе обучения. Отрицательное среднее возрастание выходов, напротив, означает, что с возрастом или во времени выход нечеткой модели уменьшается, что может быть признаком ухудшения производительности модели или снижение влияния некоторых признаков.

$$y_{mz} = \sum_{i=1}^n \omega_i \cdot y_m, \quad (15)$$

где  $n$  – число нечёткой модели,  $\omega_i$  - вес каждой нечёткой модели.

Для определения качества разработанной нечёткой модели нейронной сети рассчитываются следующие критерии:

$$E = \frac{1}{n} \sum_{i=1}^n (\hat{y}^m - y_m). \quad (16)$$

В итоге получается совокупность прогнозных моделей  $y_{mz}$  (рисунок 2).

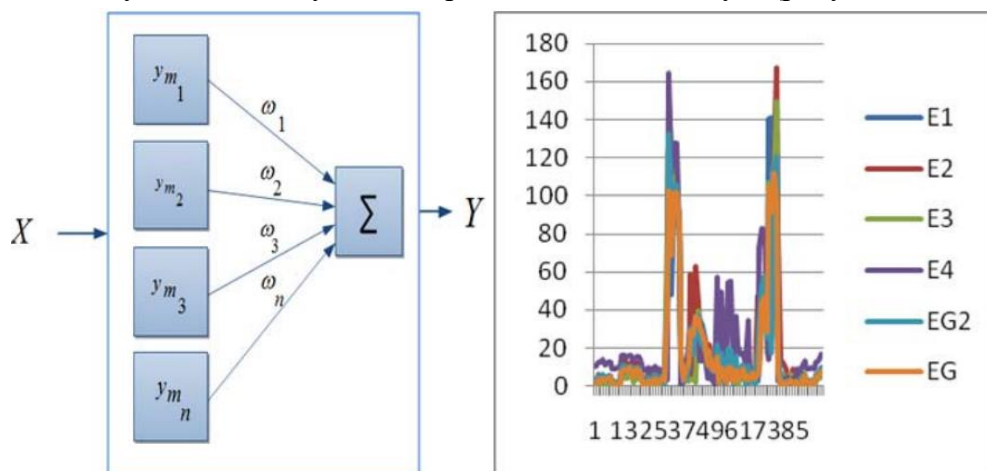


Рис 2. Схема модели (а) и график качества идентификации моделей (б)

Следовательно, получается результат определения качества идентификации модели нечёткой нейронной сети потребления электрической энергии. Средняя арифметическая ошибка составляет 1,25% (таблица 1).

Таблица 1 – Результаты качества идентификации модели нечёткой нейронной сети

| Номер эксперимента | E <sub>1</sub> | E <sub>2</sub> | E <sub>3</sub> | E <sub>4</sub> | E <sub>5</sub> | EG          |
|--------------------|----------------|----------------|----------------|----------------|----------------|-------------|
| 1                  | 1,23           | 2,01           | 2,32           | 1,25           | 1,55           | 1,22        |
| 2                  | 2,03           | 1,39           | 2,21           | 1,33           | 1,27           | 1,20        |
| 3                  | 1,36           | 0,98           | 2,56           | 1,45           | 1,45           | 1,21        |
| 4                  | 2,12           | 1,52           | 2,63           | 1,16           | 1,96           | 1,38        |
| Средняя ошибка     | 1,69           | 1,48           | 2,43           | 1,30           | 1,56           | <b>1,25</b> |

### Критерии интерпретируемости

$$K_{i^*} = (N_1 + N_2 + N_3), \quad (17)$$

$$Q_{i^*} = \left( 1 - \frac{K_i - N_{\min}}{N_{\max} - N_{\min}} \right) \cdot 100\%, \quad (18)$$

где  $N_1$  – число входных параметров в системе,  $i^*$  – индекс критерии интерпретируемости.

$$Q = \min(Q_{i^*}) \quad (19)$$

### 1.6 Применение нечётких нейронных сетей для автоматического прогнозирования потребления электрической энергии.

Для оценки точности прогнозирования суточной электрической нагрузки разработан следующий алгоритм [5]:

*Шаг № 1.* Определяется набор временных отсчётов  $T\{t_1, t_2, \dots, t_z\}$  и набор  $D$  расстояний между этими отсчётами  $T$ .

*Шаг № 2.* С помощью автокорреляционной функции определяется количество временных точек, которые могут иметь наибольшую значимость:

- 1)  $E(t - \hat{n})$  – день назад, где  $\hat{n}$  – количество временных отсчётов в течение дня;
- 2)  $E(t - 7 \cdot \hat{n})$  – количество временных отсчётов неделю назад;
- 3)  $E(t - 365 \cdot \hat{n})$  – количество временных отсчётов год назад.

*Шаг № 3.* Задаются значения входных нейронов на основе исходного временного ряда (4). Количество входных нейронов равно количеству временных отсчётов  $T = t_z$ .

$$t_z = \check{i} \cdot T, \quad (20)$$

где  $t_z$  – переменная погружения,  $\check{i}$  – индекс этой переменной.

*Шаг № 4.* С помощью формул (4)-(9) рассчитывается прогнозное значение  $\hat{y}$ .

*Шаг № 5.* Определяется качество идентификации путём вычисления ошибки на основе среднеквадратичного отклонения.

$$e^m = \frac{1}{\sum_{t=1}^T \hat{y}} \sqrt{\sum_{t=1}^T (\hat{y}(t) - y(t))^2},$$

(21)

где  $\hat{y}(t)$  – значение прогнозного потребления,  $y(t)$  – значение фактического (измеренного) потребления,  $T$  – число временных отсчётов в исследуемом интервале прогнозирования.

Для оценки точности численного прогноза суточной электрической нагрузки был проведён анализ фактического потребления электрической энергии в течении недели. Для этого использовались данные от подстанции Зыково (село в Красноярском крае) за период с



01.01.2019 по 07.01.2019. На рисунке 3 представлены реальные и прогнозные значения электроэнергии в течение этой недели.

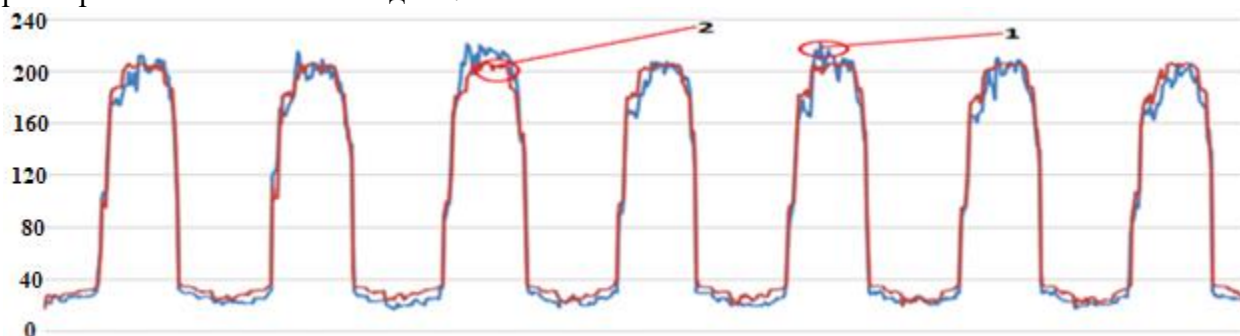


Рис. 3. График реальных (1) и прогнозных значений (2) электрической энергии в течение недели

### Заключение

В результате проведенного исследования по разработке нечетких алгоритмов для прогнозирования и потребления электрической энергии были получены следующие выводы и заключения:

1. Были разработаны и применены нечеткие алгоритмы для прогнозирования электрической энергии. Данные алгоритмы позволяют учитывать неопределенность и нечеткость данных, которые встречаются при прогнозировании электрической энергии в различных отраслях.

2. Экспериментальное исследование показало, что разработанные нечеткие алгоритмы обладают высокой точностью прогнозирования электрической энергии. Они позволяют достичь меньшей среднеквадратической ошибки прогноза по сравнению с традиционными алгоритмами.

3. Результаты исследования имеют важное практическое значение. Применение разработанных нечетких алгоритмов позволит снизить затраты на энергопотребление, улучшить энергетическую эффективность и повысить уровень экономии электрической энергии.

4. Дальнейшие исследования по данной теме могут включать расширение применения нечетких алгоритмов на другие области энергетики, разработку более точных методов прогнозирования электрической энергии и идентификацию оптимальных параметров для различных сценариев потребления электроэнергии.

5. В дальнейшей разработке по данной теме можно провести исследование применения нечетких алгоритмов на долгосрочный период энергопотребления (неделя, месяц, год и т.д.)

Таким образом, на основе проведенного исследования в данной статье можно сделать вывод, что разработка и применение нечетких алгоритмов для прогнозирования и потребления электрической энергии является эффективным и перспективным направлением исследований в области энергетики. Полученные результаты могут быть использованы в практике энергетических систем и привести к улучшению энергетической эффективности и снижению затрат на энергопотребление. Также были разработаны модели нечеткой нейронной сети для автоматического решения задачи идентификации (собственная разработка программного обеспечения автора работы). Кроме того, разработаны алгоритмы предварительной обработки информации, алгоритмы модели нечеткой нейронной сети (разработка автора статьи). В процессе работы реализованы алгоритмы в виде системы и испытаны на примере прогнозирования потребления электроэнергии.

### БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Жмак Е.И. Регулирование напряжения в электроэнергетических системах на основе нечеткой логики [Электронный ресурс]: Дис. Канд. техн. наук: 05.14.02. – М.: РГБ, 2005 (Из фондов Российской Государственной библиотеки).

2. Вятчин Д.А. Нечеткие методы автоматической классификации: Монография / Д.А. Вятчин – Мн.: УП «Технопринт», 2004 – 219 с.

3. Аль-Гунаид, М. А. Метод формирования нечетких переменных на основе временного ряда / М. А. Аль-Гунаид, М. В. Щербаков // Городу Камышину – творческую молодежь (посвящается 15-летию Камышинского технол. ин-та (филиала) ВолгГТУ): матер. III реигон. Н.-практ. студ. конф., 22–23 апр. 2009 г. / ВолГТУ, КТИ (филиал) ВолГТУ. – Камышин, 2009. – Т. 2. – С. 42–43.

4. Тарков, М. С. Нейрокомпьютерные системы // URL <http://www.intuit.ru/department/expert/neuro/15/2.html> (дата обращения: 27.05.2023).

5. Специфика применения интеллектуальных моделей анализа данных для повышения энергетической эффективности / М. В. Щербаков, Н. Л. Щербакова, Д. П. Панченко, А. Бребельс, А. П. Тюков, М. А. Аль-Гунаид // Известия ВолгГТУ : межвуз. сб. науч. ст. № 11 / ВолгГТУ. – Волгоград, 2010. – (Серия «Актуальные проблемы управления, вычислительной техники и информатики в технических системах» ; вып. 9). – С. 72–76.

## REFERENCES

1. Zhmak E.I. Voltage regulation in electric power systems based on fuzzy logic [Electronic resource]: Dis. Candidate of Technical Sciences: 05.14.02. – Moscow: RGB, 2005 (From the collections of the Russian State Library).

2. Vyatchenin D.A. Fuzzy methods of automatic classification: Monograph / D.A. Vyatchenin – Mn.: UP "Technoprint", 2004 – 219 p.

3. Al-Gunaid, M. A. Method of formation of fuzzy variables based on the time series / M. A. Al-Gunaid, M. V. Shcherbakov // The city of Kamyshin - creative youth (dedicated to the 15th anniversary of Kamyshinsky technol. in-ta (branch) VolgSTU): mater. III rejigon. N.-prakt. student. conf., 22-23 Apr. 2009 / VolGTU, KTI (branch) VolGTU. – Kamyshin, 2009. – Vol. 2. – pp. 42-43.

4. Tarkov, M. S. Neurocomputer systems // URL <http://www.intuit.ru/department/expert/neuro/15/2.html> (accessed: 05/27/2023).

5. The specifics of the use of intelligent data analysis models to improve energy efficiency / M. V. Shcherbakov, N. L. Shcherbakova, D. P. Panchenko, A. Brebels, A. P. Tyukov, M. A. Al-Gunaid // Izvestiya VolgSTU: mezhvuz. sb. scientific. art. № 11 / VolgSTU. – Volgograd, 2010. – (Series "Actual problems of management, computer engineering and informatics in technical systems" ; issue 9). – pp. 72-76.

## Информация об авторах

*Кронгауз Дмитрий Эдуардович* – к.т.н, доцент кафедры «Системы обеспечения движения поездов», Красноярский институт железнодорожного транспорта – филиал Иркутского государственного университета путей сообщения (КрИЖТ ИрГУПС), г. Красноярск, e-mail: [krongauz\\_de@krsk.irgups.ru](mailto:krongauz_de@krsk.irgups.ru)

*Рубцов Кирилл Дмитриевич* - студент 5 курса кафедры «Системы обеспечения движения поездов», Красноярский институт железнодорожного транспорта – филиал Иркутского государственного университета путей сообщения (КрИЖТ ИрГУПС), г. Красноярск, e-mail: [kirill.rubtsov.2001@mail.ru](mailto:kirill.rubtsov.2001@mail.ru).

## Information about the authors

*Krongauz Dmitry Eduardovich* – Candidate of Technical Sciences, Associate Professor of the Department of "Train Traffic Support Systems", Krasnoyarsk Institute of Railway Transport – Branch of the Irkutsk State University of Railways (KRIZHT IrGUPS), Krasnoyarsk, e-mail: [krongauz\\_de@krsk.irgups.ru](mailto:krongauz_de@krsk.irgups.ru)

*Rubtsov Kirill Dmitrievich* – 5th year student of the Department of "Train Traffic Support Systems", Krasnoyarsk Institute of Railway Transport – branch of Irkutsk State University of Railways (KRIZHT IrGUPS), Krasnoyarsk, e-mail: [kirill.rubtsov.2001@mail.ru](mailto:kirill.rubtsov.2001@mail.ru)