



doi 10.26089/NumMet.v26r317

УДК 517.912

Нейросетевой метод решения краевой задачи для дифференциальных уравнений дробного порядка

Т. Д. Нгуен

Казанский федеральный университет,
институт вычислительной математики и информационных технологий,
Казань, Российская Федерация

ORCID: 0000-0002-0993-8164, e-mail: ducnt@bcit.edu.vn

Аннотация: Многие задачи физики, механики и других наук связаны с решением краевых задач для дифференциальных уравнений дробного порядка. Найти точные решения данных задач чрезвычайно сложно, и в этом случае приходится искать приближенные решения. В настоящей работе предлагается математический метод приближенного решения краевой задачи для дифференциальных уравнений дробного порядка. Для производных дробного порядка мы используем определение подчиненной производной дробного порядка. Мы используем модель нейронной сети прямого распространения с одним скрытым слоем. Обучение модели осуществляется в режиме с учителем, при этом для оптимизации функции ошибки и коррекции параметров нейронной сети применяется алгоритм обратного распространения ошибки. Для иллюстрации нашего метода была разработана компьютерная программа для проведения экспериментов, в которых полученные результаты сравниваются с аналитическими расчетами.

Ключевые слова: дифференциальные уравнения дробного порядка, краевая задача, подчиненная производная дробного порядка, искусственная нейронная сеть, алгоритм обратного распространения ошибки.

Благодарности: Работа выполнена за счет средств Программы стратегического академического лидерства Казанского федерального университета (“ПРИОРИТЕТ-2030”).

Для цитирования: Нгуен Т.Д. Нейросетевой метод решения краевой задачи для дифференциальных уравнений дробного порядка // Вычислительные методы и программирование. 2025. 26, № 3. 245–253. doi 10.26089/NumMet.v26r317.

Neural network method for solving boundary value problems for fractional order differential equations

Tien Duc Nguyen

Kazan Federal University,
Institute of Computational Mathematics and Information Technologies,
Kazan, Russia

ORCID: 0000-0002-0993-8164, e-mail: ducnt@bcit.edu.vn

Abstract: Many problems in physics, mechanics and other sciences are related to solving boundary value problems for fractional differential equations. Finding exact solutions to these problems is very difficult, and in this case, we have to look for approximate solutions. This paper proposes a mathematical method for approximate solving of a boundary value problems for fractional differential equations. For fractional derivatives we use the definition of a conformable fractional derivative. We use a feedforward neural network model with one hidden layer. The model is trained in a supervised



learning mode using the backpropagation algorithm to optimize the error function and update the neural network parameters. To illustrate our method, a computer program was developed to conduct experiments in which the obtained results are compared with analytical solutions.

Keywords: differential equations of fractional order, boundary value problem, conformable fractional derivative, artificial neural network, error backpropagation algorithm.

Acknowledgements: This paper has been supported by the Kazan Federal University Strategic Academic Leadership Program (“PRIORITY–2030”).

For citation: T. D. Nguyen, “Neural network method for solving boundary value problems for fractional order differential equations,” Numerical Methods and Programming. 26 (3), 245–253 (2025). doi 10.26089/NumMet.v26r317.

1. Введение. Использование нейронных сетей для поиска решений дифференциальных уравнений дробного порядка является новой и необходимой областью исследований. Эти уравнения часто встречаются во многих областях науки: они возникают при описании динамических явлений в физике [1, 2], в теории электромагнитных волн [3], в биологии [4, 5], в экономике [6] и других науках [7, 8]. В частности, при изучении вязкоупругих явлений в средах возникают различные дифференциальные уравнения дробного порядка, например, такие как в моделях Скотта Блэра, Фойхта, Максвелла, Кельвина, Зенера. Существует множество работ, в которых представлены методы решения подобных дифференциальных уравнений [9–12].

В последние годы появилось много публикаций в которых нейронные сети используются для решения дифференциальных уравнений дробного порядка [13–17]. В работе [18] применялись нейронные сети и метод оптимизации Broyden–Fletcher–Goldfarb–Shanno (BFGS) для решения линейных и нелинейных дифференциальных уравнений. В работе [19] был исследован метод искусственной нейронной сети для решения дифференциальных уравнений дробного порядка с заданными начальными условиями. При этом приближенное решение было получено путем объединения начального условия с выходом нейронной сети. В статье [20] искусственные нейронные сети применялись для решения дробных линейных интегродифференциальных уравнений высшего порядка с условиями вида $y(x_1) = \beta_0$, $y'(x_2) = \beta_1$, где $y(x)$ — неизвестная функция, x_1 и x_2 — граничные точки. В статье [17] дифференциальное уравнение дробного порядка решалось с использованием нейронной сети с граничными условиями $y(0) = A$, $y(x_2) = B$. В данной работе мы решаем задачу в общем случае с граничными условиями вида $y(x_1) = A$, $y(x_2) = B$.

Во втором разделе этой работы мы даем основные определения. В разделе 3 мы предлагаем приближенный вид решения и представляем метод решения задачи с использованием нейронной сети с прямой связью. В разделе 4 мы иллюстрируем работоспособность/эффективность нашего метода на примере конкретных задач, а затем сравниваем полученные приближенные решения с точными аналитическими решениями. В заключении описаны основные результаты работы.

2. Предварительные сведения. В этом разделе мы дадим основные определения и теоремы, связанные с производными дробного порядка, которые будут использоваться в этой статье.

Определение. Подчиненной производной дробного порядка $\alpha \in (0, 1]$ функции $f : [0, \infty) \rightarrow R$ называется предел

$$T_\alpha (f(t)) = \lim_{\varepsilon \rightarrow 0} \frac{f(t + \varepsilon t^{1-\alpha}) - f(t)}{\varepsilon} \tag{1}$$

для всех $t > 0$ [9]. Функцию f в таком случае будем называть α -дифференцируемой.

Справедлива следующая теорема 1 [9]:

Теорема 1. Пусть $\alpha \in (0, 1]$, а функции f и g дифференцируемы в точке $t > 0$. Тогда

- 1) $T_\alpha (t^p) = p t^{p-\alpha}$ при $p \in R$;
- 2) для всех постоянных функций C выполняется $T_\alpha (C) = 0$;
- 3) $T_\alpha (af + bg) = aT_\alpha (f) + bT_\alpha (g)$;
- 4) $T_\alpha \left(\frac{f}{g} \right) = \frac{gT_\alpha (f) - fT_\alpha (g)}{g^2}$;



- 5) $T_\alpha(fg) = fT_\alpha(g) + gT_\alpha(f)$;
- 6) $T_\alpha(f(t)) = t^{1-\alpha} \frac{df(t)}{dt}$;
- 7) $T_\alpha(e^{bt}) = bt^{1-\alpha} e^{bt}$ npu $b \in R$;
- 8) $T_\alpha(\sin bt) = bt^{1-\alpha} \cos bt$ npu $b \in R$;
- 9) $T_\alpha(\cos bt) = -bt^{1-\alpha} \sin bt$ npu $b \in R$;
- 10) $T_\alpha\left(\frac{1}{\alpha}t^\alpha\right) = 1$.

3. Описание метода. В этом разделе мы предлагаем математический метод для нахождения приближенного решения дифференциальных уравнений дробного порядка с краевыми условиями при помощи нейронных сетей.

3.1. Основная идея. Приближенное решение $y_{NN}(x, \Omega)$ находится в виде суммы двух функций $g(x)$ и $h(x, NN(x, \Omega))$, то есть $y_{NN}(x, \Omega) = g(x) + h(x, NN(x, \Omega))$, где $g(x)$ не содержит параметров нейронной сети и выбирается так, чтобы выполнялось начальное либо граничное условие, $NN(x, \Omega)$ — выход нейронной сети, множество параметров Ω которой (веса и смещения) в ходе обучения нейросети настраивается так, чтобы средний квадрат невязки на области определения стремился к нулю. Таким образом, построенная функция удовлетворяет граничным условиям. Остается лишь настроить параметры нейронной сети так, чтобы функция $y_{NN}(x, \Omega)$ как можно лучше удовлетворяла дифференциальному уравнению на области определения, то есть чтобы максимальная норма невязки бы как можно ближе к нулю.

3.2. Описание задач и методов. Для иллюстрации метода рассмотрим дифференциальное уравнение дробного порядка вида:

$$(D^\alpha y(x)) = f(x, y), \quad x \in [x_0, x_1], \quad 0 < \alpha \leq 1 \tag{2}$$

с граничными условиями $y(x_0) = A$ и $y(x_1) = B$. Приближенное решение может быть записано в следующем виде:

$$y_{NN}(x, \Omega) = A \frac{(x_1 - x)}{(x_1 - x_0)} + B \frac{(x - x_0)}{(x_1 - x_0)} + \frac{(x_1 - x)}{(x_1 - x_0)} \frac{(x - x_0)}{(x_1 - x_0)} NN(x, \Omega). \tag{3}$$

Здесь

$$g(x) = A \frac{(x_1 - x)}{(x_1 - x_0)} + B \frac{(x - x_0)}{(x_1 - x_0)}, \quad h(x, NN(x, \Omega)) = \frac{(x_1 - x)}{(x_1 - x_0)} \frac{(x - x_0)}{(x_1 - x_0)} NN(x, \Omega).$$

При этом очевидно, что формула (3) удовлетворяет граничным условиям.

Функция ошибки $E(x, \Omega)$ определяется следующим образом:

$$E(x, \Omega) = \frac{1}{N} \sum_i \left(D^\alpha y_{NN}(x_i, \Omega) - f(x_i, y_{NN}(x_i, \Omega)) \right)^2. \tag{4}$$

Существует множество способов минимизации функции ошибки. В этой статье используется метод обратного распространения ошибки. Ниже демонстрируется как оптимизировать функцию ошибки (4) и как вычислить дробную производную $D^\alpha y_{NN}(x, \Omega)$.

3.3. Архитектура нейронной сети. На рис. 1 изображена структура нейронной сети. Пусть x — значение, поданное на вход. Оно же будет значением функции активации в первом слое.

Пусть $a_j^l(z_j^l)$ — значение функции активации j -го нейрона в l -м слое. Значение z_j^l равно выражению $z_j^l = \sum_k w_{jk}^l a_k^{l-1} + b_j^l$, где a_k^{l-1} — значение функции активации k -го нейрона в $(l - 1)$ -м слое. Значения w_{jk}^l и b_j^l — параметры весов и смещений в l -м слое. Они настраиваются в ходе

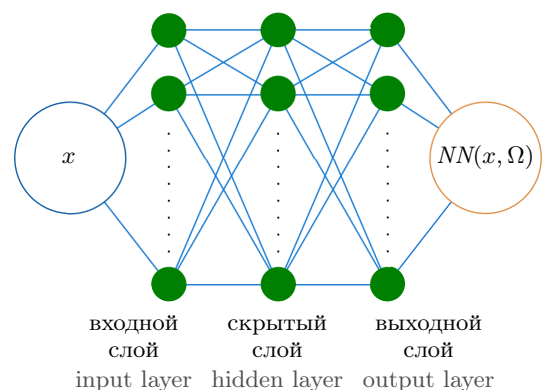


Рис. 1. Структура нейронной сети
 Fig. 1. Neural network structure

обучения нейронной сети. Входом считается значение $x \in R$, а выходом является величина

$$NN(x, \Omega) = \sum_{j=1}^m w_j^N a_j^N (z_j^{N-1}). \quad (5)$$

В данном исследовании количество скрытых слоев равно 1. На вход нейросетевой функции подается значение $x \in R$, на выходе получается $NN(x) \in R$. В качестве функции активации внутреннего слоя взят гиперболический тангенс, а в качестве алгоритма оптимизации использовался алгоритм Adam [21].

3.4. Вычисление градиентов и обновление весов. В этой статье мы использовали метод обучения в режиме с учителем, а для обновления параметров применялся метод градиентного спуска. Для обновления параметров нейронной сети и минимизации функции ошибки использовался алгоритм обратного распространения ошибки.

Обновление весов между скрытым слоем и выходным слоем осуществлялось при помощи следующей формулы:

$$v_j^{k+1} = v_j^k + \Delta v_j^k = v_j^k - \eta \frac{\partial E(x, \Omega)^k}{\partial v_j^k}, \quad (6)$$

а обновление весов между входным слоем и скрытым слоем — при помощи следующей формулы:

$$w_j^{k+1} = w_j^k + \Delta w_j^k = w_j^k - \eta \frac{\partial E(x, \Omega)^k}{\partial w_j^k}. \quad (7)$$

Здесь η — скорость обучения, k — номер итерации, используемый для обновления весов, а $E(x, \Omega)$ — функция ошибки. Для минимизации функции ошибки нам необходимо вычислить дифференциал этой функции или, другими словами, вычислить частную производную функции ошибки по параметрам.

Согласно теореме 1 выход нейронной сети рассчитывается следующим образом:

$$D_x^\alpha (NN(x, \Omega)) = w_j v_j x^{1-\alpha} \varphi'(z_j), \quad (8)$$

здесь $z_j = w_j x + b_j$, φ — функция активации.

Пусть $U = D_x^\alpha (NN(x, \Omega))$. Тогда получаем следующие частные производные:

$$\frac{\partial U}{\partial w_j} = v_j x^{1-\alpha} (\varphi'(z_j) + w_j x \varphi''(z_j)), \quad (9)$$

$$\frac{\partial U}{\partial v_j} = w_j x^{1-\alpha} \varphi'(z_j), \quad (10)$$

$$\frac{\partial U}{\partial b_j} = v_j w_j x^{1-\alpha} \varphi''(z_j). \quad (11)$$

В итоге, используя выражение (3), находим производную дробного порядка:

$$\begin{aligned} D^\alpha y_{NN}(x, \Omega) &= (B - A) \frac{x^{1-\alpha}}{(x_1 - x_0)} + \frac{x_1}{(x_1 - x_0)^2} x^{1-\alpha} NN(x, \Omega) \\ &+ \frac{x_1}{(x_1 - x_0)^2} x^{2-\alpha} NN'(x, \Omega) - \frac{x_0 x_1}{(x_1 - x_0)^2} x^{1-\alpha} NN'(x, \Omega) \\ &- 2 \frac{x^{2-\alpha}}{(x_1 - x_0)^2} NN(x, \Omega) - \frac{x^{3-\alpha}}{(x_1 - x_0)^2} NN'(x, \Omega) \\ &+ \frac{x_0}{(x_1 - x_0)^2} x^{1-\alpha} NN(x, \Omega) + \frac{x_0}{(x_1 - x_0)^2} x^{2-\alpha} NN'(x, \Omega). \end{aligned} \quad (12)$$

4. Вычислительный эксперимент. В этом разделе мы продемонстрируем работу наших методов на конкретных примерах. Также используем наш метод для приближенного решения задачи Дирихле, где граничные условия произвольны. Для разработки программы использовалась библиотека глубокого обучения PyTorch, после чего были проведены серии экспериментов. Мы используем нейронную сеть с одним скрытым слоем. Алгоритм обратного распространения ошибки используется для оптимизации функции



ошибки и обновления параметров. В качестве функции активации для скрытого слоя нейронной сети берется гиперболический тангенс $\phi(x) = \tanh(x)$. Программа тестируется на компьютере с процессором 13th Gen Intel(R) Core i7-13700 (24 ядра), тактовой частотой 2,1 ГГц и оперативной памятью 16384 МБ.

Эксперимент 1. Рассмотрим следующее дифференциальное уравнение дробного порядка

$$D_x^{0.75}y(x) - x^{1/4}y(x) = x^{1/4}e^x \cos x, \quad 0 \leq x \leq \frac{\pi}{2} \quad (13)$$

с граничными условиями $y(0) = 0, y\left(\frac{\pi}{2}\right) = e^{\pi/2}$.

Это уравнение имеет аналитическое решение $y(x) = e^x \sin x$.

Используя формулу (3), находим следующее выражение для приближенного решения, полученного с помощью нейронной сети:

$$y_{NN}(x, \Omega) = \frac{2x}{\pi} \left(e^{\pi/2} + \frac{\pi - 2x}{\pi} NN(x, \Omega) \right).$$

Мы обучаем сеть на десяти точках из интервала $\left[0; \frac{\pi}{2}\right]$. Архитектура сети включает один скрытый слой с десятью нейронами. Обучение выполняется в течение 20 эпох. Скорость обучения $\eta = 0.1$. Различие и невязка между аналитическим решением и приближенным решением, полученным при помощи нейронной сети, показаны на рис. 2, 3. Время выполнения составляет около 2.1 с.

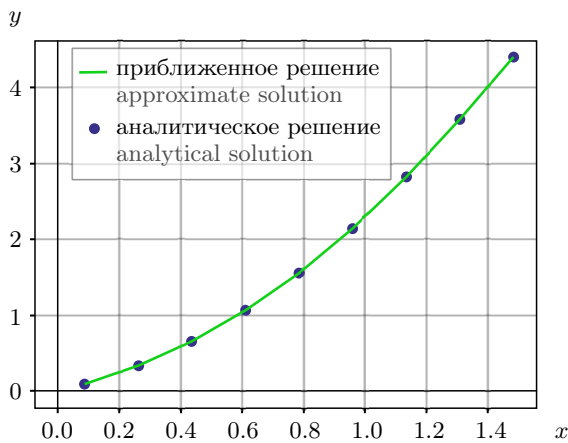


Рис. 2. Сравнение поведения аналитического и приближенного решений

Fig. 2. Comparison of the behavior of analytical and approximate solutions

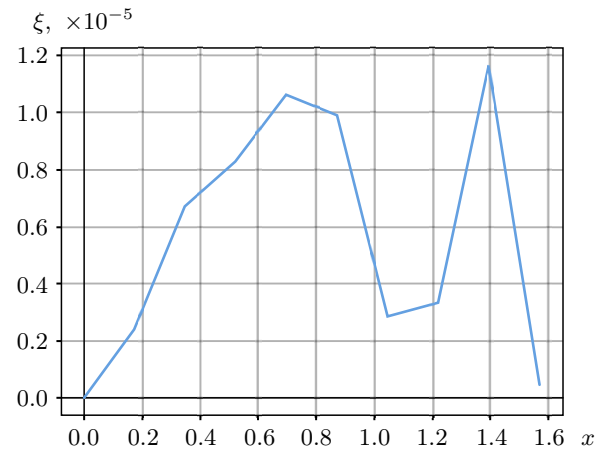


Рис. 3. Невязка ξ между аналитическим и приближенным решением уравнения (13)

Fig. 3. Residual error ξ between the analytical and approximate solutions of equation (13)

Эксперимент 2. Рассмотрим следующее дифференциальное уравнение задачи Дирихле

$$y''(x) - x^2y(x) + D_x^{0.5}y(x) = -\left(\frac{\pi^2}{4} + x^2\right) \cos\left(\frac{\pi x}{2}\right) - \frac{\pi}{2}\sqrt{x} \sin\left(\frac{\pi x}{2}\right), \quad \frac{1}{2} \leq x \leq 2 \quad (14)$$

с граничными условиями $y\left(\frac{1}{2}\right) = \frac{\sqrt{2}}{2}, y(2) = -1$.

Это уравнение имеет аналитическое решение $y(x) = \cos\left(\frac{\pi x}{2}\right)$.

Выражение для приближенного решения уравнения (14), найденного при помощи нейронной сети, выглядит следующим образом:

$$y_{NN}(x, \Omega) = \frac{\sqrt{2}(2-x)}{3} + \frac{2x-1}{3} + \frac{2(2-x)(2x-1)}{9} NN(x, \Omega).$$

Мы обучаем нейронную сеть на двадцати точках из области $\left[\frac{1}{2}; 2\right]$. Архитектура сети включает один скрытый слой с десятью нейронами. Обучение выполняется в течение 20 эпох. Скорость обучения

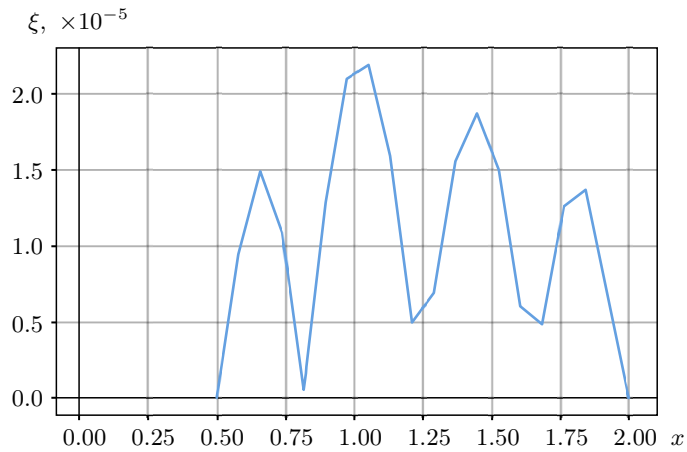


Рис. 4. Невязка ξ между аналитическим и приближенным решением уравнения (14)

Fig. 4. Residual error ξ between the analytical and approximate solutions of equation (14)

$\eta = 0.1$. Невязка между аналитическим решением и приближенным решением, полученным при помощи нейронной сети, показана на рис. 4. Время выполнения составляет около 2.21 с.

Эксперимент 3: Теперь рассмотрим следующее дифференциальное уравнение дробного порядка:

$$D_x^{0.75}y(x) = 2x^{0.25}(1 - 2\sin^2 x), \quad \frac{\pi}{4} \leq x \leq \pi \tag{15}$$

с граничными условиями $y\left(\frac{\pi}{4}\right) = 1, y(\pi) = 0$.

Это уравнение имеет аналитическое решение $y(x) = \sin 2x$.

Выражение для приближенного решения уравнения (15), найденного при помощи нейронной сети, выглядит следующим образом:

$$y_{NN}(x, \Omega) = \frac{4(\pi - x)}{3\pi} \left(1 + \frac{(4x - \pi)}{3\pi} NN(x, \Omega) \right).$$

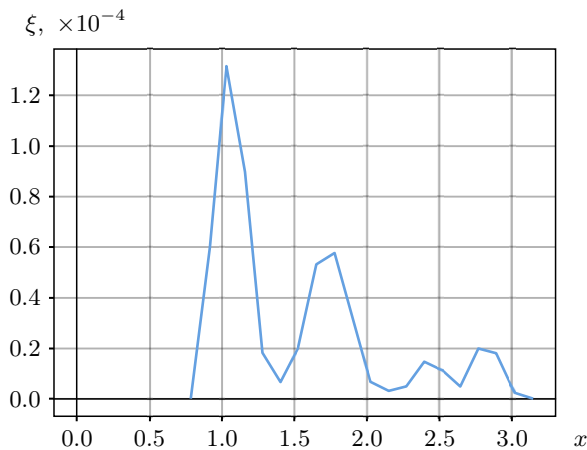


Рис. 5. Невязка ξ между аналитическим и приближенным решением уравнения (15) при использовании нейронной сети с 5 нейронами в скрытом слое

Fig. 5. Residual error ξ between the analytical and approximate solutions of equation (15) when using a neural network with 5 neurons in the hidden layer

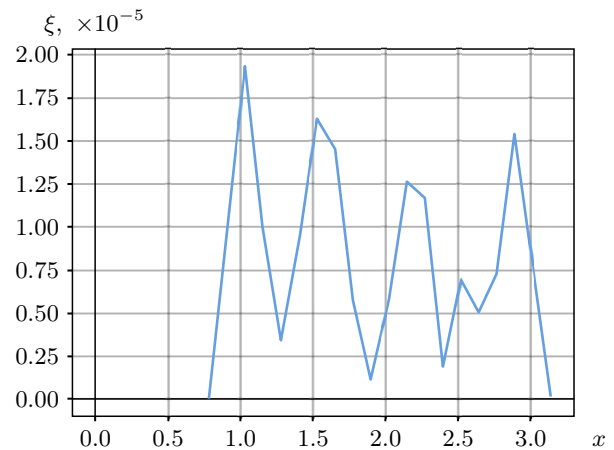


Рис. 6. Невязка ξ между аналитическим и приближенным решением уравнения (15) при использовании нейронной сети с 10 нейронами в скрытом слое

Fig. 6. Residual error ξ between the analytical and approximate solutions of equation (15) when using a neural network with 10 neurons in the hidden layer



Мы обучаем сеть на двадцати точках из области $\left[\frac{\pi}{4}; \pi\right]$. Архитектура сети включает один скрытый слой с пятью и десятью нейронами. Обучение выполняется в течение 30 эпох. Скорость обучения $\eta = 0.1$. Невязка между аналитическим решением и приближенным решением, полученным при помощи нейронной сети, показана на рис. 5, 6. Время выполнения составляет около 2.41 с.

Из полученных изображений видно, что наш метод очень хорошо аппроксимирует решения дифференциальных уравнений дробного порядка. При использовании нейронных сетей для решения дифференциальных уравнений часто возникают трудности с определением количества скрытых слоев, числа нейронов в каждом скрытом слое, скорости обучения и т.д. В данной статье выбор параметров был проведен экспериментально, а их конкретные значения обсуждаются ниже.

5. Заключение. При использовании нейронных сетей для решения дифференциальных уравнений дробного порядка основная сложность состоит в том, как построить формулу приближенного решения, а также в том, как вычислить производные дробного порядка для функции ошибки. В данной работе мы предлагаем математический метод для решения краевой задачи в общем виде. Для этого мы используем понятие подчиненной производной дробного порядка, которое позволяет преобразовать дробную производную в целочисленную. Нами были проведены эксперименты с разной скоростью обучения, а также с разными эпохами и количеством нейронов в скрытом слое. Результаты предложенного метода сравниваются с результатами аналитических расчетов. На основании полученных результатов мы делаем следующий вывод: метод сходится уже после 10–15 итераций, приближенные решения хорошо аппроксимируют точные решения (ошибки не превосходят 10^{-7} – 10^{-4}), метод требует малого времени выполнения и небольшого количества нейронов (около 10 нейронов). Результаты экспериментов показывают, что предлагаемый нами метод эффективно работает, является простым и имеет высокую вычислительную эффективность.

Список литературы

1. *Verhulst P.F.* Resherches mathématiques sur la loi d'accroissement de la population // Nouveaux Mémoires de l'Academie Royale des Sciences. 1845. **18**. 14–54. <https://eudml.org/doc/182533>. Cited June 5, 2025.
2. *Xu M., Tan W.* Intermediate processes and critical phenomena: theory, method and progress of fractional operators and their applications to modern mechanics // Science in China, Series G. 2006. **49**, N 3. 257–272. doi [10.1007/s11433-006-0257-2](https://doi.org/10.1007/s11433-006-0257-2).
3. *Malthus T.R.* An essay on the principle of population. London: J. Johnson, 1798.
4. *Li K., Wei J.* Stability and Hopf bifurcation analysis of a prey–predator system with two delays // Chaos, Solitons & Fractals. 2009. **42**, N 5. 2606–2613. doi [10.1016/j.chaos.2009.04.001](https://doi.org/10.1016/j.chaos.2009.04.001).
5. *Mackey M.C., Glass L.* Oscillation and chaos in physiological control systems // Science. 1977. **197**, N 4300. 287–289. doi [10.1126/science.267326](https://doi.org/10.1126/science.267326).
6. *Chen W.-C.* Nonlinear dynamics and chaos in a fractional-order financial system // Chaos, Solitons & Fractals. 2008. **36**, N 5. 1305–1314. doi [10.1016/j.chaos.2006.07.051](https://doi.org/10.1016/j.chaos.2006.07.051).
7. *Chen F., Zhou L.* Strange attractors in a periodically perturbed Lorenz-like equation // J. Appl. Anal. Comput. 2013. **3**, N 2. 123–132. doi [10.11948/2013010](https://doi.org/10.11948/2013010).
8. *Yang Z., Jiang T., Jing Z.* Bifurcations of periodic solutions and chaos in Duffing–Van der Pol equation with one external forcing // J. Appl. Anal. Comput. 2013. **3**, N 4. 405–431. doi [10.11948/2013030](https://doi.org/10.11948/2013030).
9. *Ünal E., Gökdoğan A.* Solution of conformable fractional ordinary differential equations via differential transform method // Optik. 2017. **128**. 264–273. doi [10.1016/j.ijleo.2016.10.031](https://doi.org/10.1016/j.ijleo.2016.10.031).
10. *Miller K.S., Ross B.* An introduction to the fractional calculus and fractional differential equations. New York: Wiley, 1993.
11. *Merdan M.* On the solutions of fractional Riccati differential equation with modified Riemann–Liouville derivative // Int. J. Differ. Equat. 2012. **2012**, Article ID 346089. doi [10.1155/2012/346089](https://doi.org/10.1155/2012/346089).
12. *Demirci E., Ozalp N.* A method for solving differential equations of fractional order // J. Comput. Appl. Math. 2012. **236**, N 11. 2754–2762. doi [10.1016/j.cam.2012.01.005](https://doi.org/10.1016/j.cam.2012.01.005).
13. *Khalil R., Al Horani M., Yousef A., Sababheh M.* A new definition of fractional derivative // J. Comput. Appl. Math. 2014. **264**. 65–70. doi [10.1016/j.cam.2014.01.002](https://doi.org/10.1016/j.cam.2014.01.002).
14. *Khan N.A., Ara A., Khan N.A.* Fractional-order Riccati differential equation: analytical approximation and numerical results // Adv. Differ. Equ. 2013. **2013**, Article Number: 185. doi [10.1186/1687-1847-2013-185](https://doi.org/10.1186/1687-1847-2013-185).
15. *Duc N.T., Galimyanov A.F., Akhmetov I.Z.* Artificial neural network method for solving a fractional order differential equation with the Cauchy-type problem // Proc. 2023 Int. Russian Smart Industry Conf. Sochi, 2023. 329–334. doi [10.1109/SmartIndustryCon57312.2023.10110823](https://doi.org/10.1109/SmartIndustryCon57312.2023.10110823).

16. Duc N.T., Galimyanov A.F., Akhmetov I.Z. Neural network method for solving fractional differential equations α with the Dirichlet problem // Proc. 2023 Int. Russian Smart Industry Conf. Sochi, 2023. 295–300. doi [10.1109/SmartIndustryCon57312.2023.10110785](https://doi.org/10.1109/SmartIndustryCon57312.2023.10110785).
17. Gao F., Dong Yu., Chi Ch. Solving fractional differential equations by using triangle neural network // J. Funct. Spaces. 2021. **2021**, Article ID 5589905. doi [10.1155/2021/5589905](https://doi.org/10.1155/2021/5589905).
18. Pakdaman M., Ahmadian A., Effati S., et al. Solving differential equations of fractional order using an optimization technique based on training artificial neural network // Appl. Math. Comput. 2017. **293**. 81–95. doi [10.1016/j.amc.2016.07.021](https://doi.org/10.1016/j.amc.2016.07.021).
19. Mall S., Chakraverty S. Artificial neural network approach for solving fractional order initial value problems. <https://arxiv.org/abs/1810.04992>. Cited June 7, 2025.
20. Allahviranloo T., Jafarian A., Saneifard R., et al. An application of artificial neural networks for solving fractional higher-order linear integro-differential equations // Bound. Value Probl. 2023. **2023**, Article Number: 74. doi [10.1186/s13661-023-01762-x](https://doi.org/10.1186/s13661-023-01762-x).
21. Хайкин С. Нейронные сети: полный курс. М.: Вильямс, 2019.

Получена
5 сентября 2024 г.

Принята
2 июня 2025 г.

Опубликована
11 июля 2025 г.

Информация об авторе

Тиен Дык Нгуен — аспирант, Казанский федеральный университет, институт вычислительной математики и информационных технологий, ул. Кремлевская, 35, 420008, Казань, Российская Федерация.

References

1. P. F. Verhulst, “Resherches mathématiques sur la loi d’accroissement de la population,” *Nouveaux Mémoires de l’Academie Royale des Sciences* **18**, 14–54 (1845). <https://eudml.org/doc/182533>. Cited June 5, 2025.
2. M. Xu and W. Tan, “Intermediate Processes and Critical Phenomena: Theory, Method and Progress of Fractional Operators and Their Applications to Modern Mechanics,” *Sci. China Ser. G* **49** (3), 257–272 (2006). doi [10.1007/s11433-006-0257-2](https://doi.org/10.1007/s11433-006-0257-2).
3. T. R. Malthus, *An Essay on the Principle of Population* (J. Johnson, London, 1798).
4. K. Li and J. Wei, “Stability and Hopf Bifurcation Analysis of a Prey–Predator System with Two Delays,” *Chaos Solit. Fractals* **42** (5), 2606–2613 (2009). doi [10.1016/j.chaos.2009.04.001](https://doi.org/10.1016/j.chaos.2009.04.001).
5. M. C. Mackey and L. Glass, “Oscillation and Chaos in Physiological Control Systems,” *Science* **197** (4300), 287–289 (1977). doi [10.1126/science.267326](https://doi.org/10.1126/science.267326).
6. W.-C. Chen, “Nonlinear Dynamics and Chaos in a Fractional-Order Financial System,” *Chaos Solit. Fractals* **36** (5), 1305–1314 (2008). doi [10.1016/j.chaos.2006.07.051](https://doi.org/10.1016/j.chaos.2006.07.051).
7. F. Chen and L. Zhou, “Strange Attractors in a Periodically Perturbed Lorenz-like Equation,” *J. Appl. Anal. Comput.* **3** (2), 123–132 (2013). doi [10.11948/2013010](https://doi.org/10.11948/2013010).
8. Z. Yang, T. Jiang, and Z. Jing, “Bifurcations of Periodic Solutions and Chaos in Duffing–Van der Pol Equation with One External Forcing,” *J. Appl. Anal. Comput.* **3** (4), 405–431 (2013). doi [10.11948/2013030](https://doi.org/10.11948/2013030).
9. E. Ünal and A. Gökdoğan, “Solution of Conformable Fractional Ordinary Differential Equations via Differential Transform Method,” *Optik* **128**, 264–273 (2017). doi [10.1016/j.ijleo.2016.10.031](https://doi.org/10.1016/j.ijleo.2016.10.031).
10. K. S. Miller and B. Ross, *An Introduction to the Fractional Calculus and Fractional Differential Equations* (Wiley, New York, 1993).
11. M. Merdan, “On the Solutions of Fractional Riccati Differential Equation with Modified Riemann–Liouville Derivative,” *Int. J. Differ. Equat.* **2012**, Article ID 346089 (2012). doi [10.1155/2012/346089](https://doi.org/10.1155/2012/346089).
12. E. Demirci and N. Ozalp, “A Method for Solving Differential Equations of Fractional Order,” *J. Comput. Appl. Math.* **236** (11), 2754–2762 (2012). doi [10.1016/j.cam.2012.01.005](https://doi.org/10.1016/j.cam.2012.01.005).
13. R. Khalil, M. Al Horani, A. Yousef, and M. Sababheh, “A New Definition of Fractional Derivative,” *J. Comput. Appl. Math.* **264**, 65–70 (2014). doi [10.1016/j.cam.2014.01.002](https://doi.org/10.1016/j.cam.2014.01.002).
14. N. A. Khan, A. Ara, and N. A. Khan, “Fractional-Order Riccati Differential Equation: Analytical Approximation and Numerical Results,” *Adv. Differ. Equ.* **2013**, Article Number: 185 (2013). doi [10.1186/1687-1847-2013-185](https://doi.org/10.1186/1687-1847-2013-185).



15. N. T. Duc, A. F. Galimyanov, and I. Z. Akhmetov, “Artificial Neural Network Method for Solving a Fractional Order Differential Equation with the Cauchy-Type Problem,” in *Proc. 2023 Int. Russian Smart Industry Conf. (SmartIndustryCon), Sochi, Russian Federation, March 27–31, pp. 329–334* (2023). doi [10.1109/SmartIndustryCon57312.2023.10110823](https://doi.org/10.1109/SmartIndustryCon57312.2023.10110823).
16. N. T. Duc, A. F. Galimyanov, and I. Z. Akhmetov, “Neural Network Method for Solving Fractional Differential Equations α with the Dirichlet Problem,” in *Proc. 2023 Int. Russian Smart Industry Conf. (SmartIndustryCon), Sochi, Russian Federation, March 27–31, pp. 295–300* (2023). doi [10.1109/SmartIndustryCon57312.2023.10110785](https://doi.org/10.1109/SmartIndustryCon57312.2023.10110785).
17. F. Gao, Yu. Dong, and Ch. Chi, “Solving Fractional Differential Equations by Using Triangle Neural Network,” *J. Funct. Spaces* **2021**, Article ID 5589905 (2021). doi [10.1155/2021/5589905](https://doi.org/10.1155/2021/5589905).
18. M. Pakdaman, A. Ahmadian, S. Effati, et al., “Solving Differential Equations of Fractional Order Using an Optimization Technique Based on Training Artificial Neural Network,” *Appl. Math. Comput.* **293**, 81–95 (2017). doi [10.1016/j.amc.2016.07.021](https://doi.org/10.1016/j.amc.2016.07.021).
19. S. Mall and S. Chakraverty, “Artificial Neural Network Approach for Solving Fractional Order Initial Value Problems,” <https://arxiv.org/abs/1810.04992>. Cited June 7, 2025.
20. T. Allahviranloo, A. Jafarian, R. Saneifard, et al., “An Application of Artificial Neural Networks for Solving Fractional Higher-Order Linear Integro-Differential Equations,” *Bound. Value Probl.* **2023**, Article Number: 74 (2023). doi [10.1186/s13661-023-01762-x](https://doi.org/10.1186/s13661-023-01762-x).
21. S. Haykin, *Neural Networks: A Comprehensive Foundation* (Prentice Hall, Upper Saddle River, 1999).

Received
September 5, 2024

Accepted
June 2, 2025

Published
July 11, 2025

Information about the author

Tien Duc Nguyen — Graduate Student, Kazan Federal University, Institute of Computational Mathematics and Information Technologies, Kremlevskaya ulitsa, 35, 420008, Kazan, Russia.