# О ВОССТАНОВЛЕНИИ КУЛОНОВСКОГО ПОЛЯ ПОТЕНЦИАЛЬНЫХ ТЕЧЕНИЙ НА ОСНОВЕ ИСКУССТВЕННЫХ НЕЙРОННЫХ СЕТЕЙ

**А. Д. Смородинов**[1,2]

[1] *Сургутский филиал федерального государственного автономного учреждения «Федеральный научный центр Научно-исследовательский институт системных исследований Национального исследовательского центра «Курчатовский институт», г. Сургут, Российская Федерация*

[2] *Сургутский государственный университет, г. Сургут, Российская Федерация*

*ORCID: https://orcid.org/0000-0002-9324-1844,* ✍ *smorodinov.alexsandr@yandex.ru*

*Аннотация*: в статье рассматривается возможность применения искусственной нейронной сети для решения задачи Коши на примере моделирования течения тяжелой примеси в потенциальных течениях несжимаемой жидкости, порожденных гармоническим потенциалом. Конструируется искусственная нейронная сеть, не требующая обучения, для представления градиента кулоновского потенциала, показывающая высокую точность приближения. Показывается связь итерационных методов решения задачи Коши с ИНС, конструируется ИНС, реализующая метод Рунге–Кутты, в частности, 4-го порядка точности, не требующая обучения, в которой весовые коэффициенты установлены на этапе конструирования. Проводится тестовый расчет, в котором сравниваются результаты моделирования динамики примеси с использованием разработанной ИНС и программного комплекса для решения задачи Коши с различными конфигурациями полей начальных данных и визуализации результатов математического моделирования течения тяжелой примеси. Кроме того, показывается возможность практического применения ИНС для поиска местоположений источника кулоновского потенциала.

*Ключевые слова*: искусственные нейронные сети, теорема Колмогорова-Арнольда, задача Коши, метод Рунге–Кутты, динамика примеси.

*Благодарности*: работа выполнена в рамках государственного задания НИЦ «Курчатовский институт» — НИИСИ по теме № FNEF-2024-0001 «Создание и реализация доверенных систем искусственного интеллекта, основанных на новых математических и алгоритмических методах, моделях быстрых вычислений, реализуемых на отечественных вычислительных системах» (1023032100070-3-1.2.1).

*Для цитирования*: Смородинов А. Д. О восстановлении кулоновского поля потенциальных течений на основе искусственных нейронных сетей. *Успехи кибернетики*. 2025;6(1):137–149.

*Поступила в редакцию: 27.02.2025.* *В окончательном варианте: 09.03.2025.*

# NEURAL NETWORK APPLICATIONS TO THE RECONSTRUCTION OF COULOMB FIELD IN POTENTIAL FLOWS

**A. D. Smorodinov**[1,2]

[1] *Surgut Branch of Scientific Research Institute for System Analysis of the National Research Centre "Kurchatov Institute", Surgut, Russian Federation*

[2] *Surgut State University, Surgut, Russian Federation*

*ORCID: https://orcid.org/0000-0002-9324-1844,* ✍ *smorodinov.alexsandr@yandex.ru*

*Abstract*: we apply an artificial neural network (ANN) to solve the Cauchy problem by modeling the flow of heavy impurities in potential flows of an incompressible fluid generated by a harmonic potential. We construct an ANN that does not require training to represent the gradient of the Coulomb potential, demonstrating high approximation accuracy. We establish the connection between iterative methods for solving the Cauchy problem and ANNs. Specifically, we design an ANN that implements the Runge-Kutta method, including a fourth-order accurate version, where weight coefficients are predefined at the design stage, eliminating the need for training. We perform a test calculation to compare impurity dynamics simulated using the developed ANN with results from a software package for solving the Cauchy problem. We analyze different initial data configurations and visualize the simulated flow of heavy impurities. Additionally, we demonstrate the practical application of the ANN for locating sources of the Coulomb potential.

*Keywords*: artificial neural networks, Kolmogorov-Arnold theorem, Cauchy problem, Runge-Kutta method, impurity dynamics.

*Acknowledgements*: this is a part of the FNEF-2024-0001 Development and Deployment of Trusted AI Systems based on New Mathematical and Algorithmic Approaches and Fast Computing Models Compatible with Domestic Computer Hardware government contract (1023032100070-3-1.2.1) granted to the NRC "Kurchatov Institute" – SRISA.

*Cite this article*: Smorodinov A. D. Neural Network Applications to the Reconstruction of Coulomb Field in Potential Flows. *Russian Journal of Cybernetics*. 2025;6(1):137–149.

*Original article submitted: 27.02.2025.* *Revision submitted: 09.03.2025.*

**Введение**

Искусственные нейронные сети (ИНС) в настоящее время показывают хорошие результаты при решении некоторых задач. Это обусловлено тем, что часто для разработки систем искусственного интеллекта используют предобученные ИНС, в которых уже были установлены функции активации, размерность ИНС и определены основные значения весовых коэффициентов. В таких случаях исследователю остается только дообучить ИНС на своем наборе данных для решения более узкоспециализированной задачи.

В данной работе представляется предварительно сконструированная ИНС, весовые коэффициенты которой установлены исходя из аналитических соображений, основанных на итерационном методе решения задачи Коши. Такой подход может быть использован для решения задачи, которая возникает у экологической полиции: предположим, что есть информация о загрязнении некоторого водоема вследствие сброса загрязняющих веществ в воду. Необходимо, имея информацию о движении загрязнений (примеси) в некоторой части водоема, найти местоположение и силу сброса. Покажем создание описанной ИНС, а также применение ИНС для решения задачи. По сути, будет сконструирована предобученная ИНС, которую необходимо будет дообучить для определения местоположения источников на конкретной задаче.

**Постановка задачи математического моделирования**

Рассмотрим возможность применения ИНС к задаче гидродинамики на примере задачи моделирования движения безынерционных недиффундирующих примесей. Для этого рассмотрим следующее поле скоростей (1), которое является точным решением системы уравнений Навье–Стокса:

$$\vec{V} = \nabla\varphi\left(\vec{x},t\right), \tag{1}$$

где $\varphi\left(\vec{x},t\right)$ — скалярное поле, которое удовлетворяет уравнению Лапласа:

$$\Delta_x\varphi(\vec{x},t) \equiv 0. \tag{2}$$

Выберем в качестве скалярного поля поле следующего вида:

$$\varphi\left(\vec{x},t\right) = \frac{q_i}{\|\vec{x} - M_1\|}, \tag{3}$$

$$\vec{V}\left(\vec{x}\right) = \nabla\left(\sum_{i=1}^{K}\frac{q_i}{\|\vec{x} - M_i\|}\right), \tag{4}$$

где знаменатель есть евклидова норма:

$$\|\vec{x} - M_1\| = \sqrt{\left(x_1 - M_1^1\right)^2 + \left(x_2 - M_1^2\right)^2 + \left(x_3 - M_1^3\right)^2}, \tag{5}$$

где $K$ — количество потенциальных полей, $q_i$ — мощность i-го источника, $M_i$ — местоположение i-го источника.

Тогда моделирование динамики примеси, движущейся в потенциальных течениях, задаваемых кулоновскими потенциалами, описывается уравнением:

$$\frac{\partial N}{\partial t} + \left(\vec{V}\cdot\nabla\right)N = \sigma^2\cdot\Delta N, \tag{6}$$

где $N$ – концентрация примеси, т.к. рассматривается движение безынерционных недиффундирующих примесей, то коэффициент диффузии $\sigma = 0$, а вектор скорости движения примеси совпадает с вектором скорости движения жидкости $\vec{V}(\vec{x})$, и тогда уравнение (6) примет следующий вид:

$$\frac{\partial N}{\partial t} = -\left(\vec{V}\cdot\nabla\right)N. \tag{7}$$

Следовательно, для получения траекторий движения частиц примеси необходимо решить задачу Коши в трехмерном пространстве.

В статье [1] приведена полная постановка задачи, а также исследована структура течений и представлена серия вычислительных экспериментов по моделированию динамики примеси во встречных потенциальных потоках одинаковой мощности, а также сравнение с натурным экспериментом столкновения двух встречных струй с использованием программного обеспечения [17], описанного в [2]. Покажем, что данная задача может быть решена на основе ИНС.

**Конструирование ИНС для решения задачи Коши**

Обратимся к теореме Колмогорова–Арнольда о суперпозиции функций многих переменных функцией одной переменной [3].

Теорема Колмогорова–Арнольда: при любом целом $n\geq 2$ существуют такие определенные на единичном отрезке $E^1 = [0;1]$ непрерывные действительные функции $\psi^{pq}(x)$, что каждая определенная на $n$-мерном единичном кубе $E^n$ непрерывная действительная функция $f(x_1,\ldots,x_n)$ представима в виде:

$$f(x_1,\ldots,x_n) = \sum_{q=1}^{q=2n+1} \chi_q\left[\sum_{p=1}^{n}\psi^{pq}(x_p)\right], \tag{8}$$

где функции $\chi_q(y)$ действительны и непрерывны.

Фактически в данной теореме показано, что любую функцию многих переменных можно представить в виде суперпозиции функции одной переменной и сложения. В более поздних работах показано, что внутренние функции $\psi$ можно заменить одной [5]. Применительно к ИНС похожая модификация теоремы предложена в работе [6]. Строго говоря, теоремы, представленные в работах [5, 6], не повторяют теорему Колмогорова-Арнольда в точности в отношении суперпозиции, а говорят, что такие функции можно подобрать в зависимости от наперед заданной точности приближения.

На данную теорему можно смотреть с точки зрения расширения теоремы Вейерштрасса [4], в которой показано, что любая непрерывная функция на некотором замкнутом интервале действительных чисел может быть представлена абсолютно и равномерно сходящимся рядом полиномов.

Данная теорема является теоремой существования, из которой, строго говоря, не следует, что данная конфигурация является оптимальной с точки зрения времени обучения или что используемые функции являются легко реализуемыми, поэтому чаще всего для решения подобных задач применяются многослойные перцептроны с большим количеством нейронов в каждом слое. В связи с этим при конструировании ИНС будем опираться на данную теорему, но не будем требовать полного совпадения структуры полученной ИНС с теоремой.

Покажем, что поле скоростей (4) представимо в виде суперпозиции функций одной переменной и может быть реализовано как ИНС. Для этого распишем оператор набла в (4) и получим поле $\vec{V}(\vec{x})$ в покомпонентной записи:

$$V_i(x_1,x_2,x_3) = -\sum_{k=1}^{K}\left[q_k\left\{\left(x_1 - M_k^1\right)^2 + \left(x_2 - M_k^2\right)^2 + \left(x_3 - M_k^3\right)^2\right\}^{-\frac{1}{3}}(x_i - M_k^i)\right], \tag{9}$$

где $i = 1,\ldots,3$ — количество компонент, $M_k^i$ — i-ая координата k-го источника.

Покажем, что (9) представимо в виде классической многослойной ИНС, особенностью которой будет отсутствие необходимости проводить поиск весовых коэффициентов сети. Фактически в данной задаче искусственная нейронная сеть будет представлять собой вычислительный ориентированный ациклический граф.

Для этого введем следующие обозначения для элементарных функций:

$$G^2(x) = x^2, \tag{10}$$

$$G^{-\frac{3}{2}}(x) = x^{-\frac{3}{2}}, \tag{11}$$

$$G(x) = x. \tag{12}$$

Формулы (10)–(12) будут являться функциями активации, а константы в формуле (8) будут весовыми коэффициентами. Тогда формулы (9) преобразуются в следующий вид:

$$V_i(x_1,x_2,x_3) = -\sum_{k=1}^{K}\left[q_k G^{-\frac{3}{2}}\left\{G^2\left(x_1 - M_k^1\right) + G^2\left(x_2 - M_k^2\right) + G^2\left(x_3 - M_k^3\right)\right\} G\left(x_i - M_k^i\right)\right]. \tag{13}$$

Упростим запись формул (13) и аргумент функции $\mathrm{G}^{-\frac{3}{2}}$ запишем под знаком суммы, тогда $V_i$ будет иметь следующую запись:

$$V_i(x_1,x_2,x_3) = -\sum_{k=1}^{K}\left[q_k G^{-\frac{3}{2}}\left\{\sum_{j=1}^{3} G^2(x_j - M_k^j)\right\} G(x_i - M_k^i)\right]. \tag{14}$$

В классических ИНС нет умножения функций, а только сложение, потому распишем умножение функции $G^{-\frac{3}{2}}$ на $G$, используя следующую формулу, основанную на формуле сокращенного умножения:

$$x_1 x_2 = \frac{(x_1 + x_2)^2 - x_1^2 - x_2^2}{2}. \tag{15}$$

Заметим, что данная формула легко представима с помощью функций (10) и (12):

$$x_1 \cdot x_2 = \frac{G^2\left(\sum\limits_{m=1}^{2} G(x_m)\right) - \sum\limits_{m=1}^{2} G^2(x_m)}{2}. \tag{16}$$

Тогда, подставив вместо $x_1$ функцию $G^{-\frac{3}{2}}$, вместо $x_2$ функцию $G$ и результат в (9), вместо умножения соответствующих формул получаем:

$$V_i(x_1,x_2,x_3) = -\sum_{k=1}^{K}\left[q_k \frac{G^2\left(G^{-\frac{3}{2}}\left\{\sum\limits_{j=1}^{3} G^2\left(x_j - M_k^j\right)\right\} + G\left(x_i - M_k^i\right)\right) - G^2\left(G^{-\frac{3}{2}}\left\{\sum\limits_{j=1}^{3} G^2\left(x_j - M_k^j\right)\right\}\right) - G^2\left(G\left(x_i - M_k^i\right)\right)}{2}\right]. \tag{17}$$

Данная формула показывает, что (9) представима в виде ИНС.

Примем гипотезу из некоторой априорной информации о кулоновском виде потенциальных течений в виде суперпозиции конечного числа точечных источников и стоков.

В несколько этапов спроектируем данную ИНС. Для начала сконструируем ИНС для умножения двух чисел. Особенностью данной ИНС является ее сложная, составная структура, которую можно представить в виде вычислительного ациклического графа посредством библиотеки для реализации ИНС Keras для языка программирования Python.

**Модель искусственной нейронной сети для умножения двух чисел**

Покажем реализацию ИНС для сети (17) с реализацией части, отвечающей за умножение двух чисел. Данная модель ИНС описывается равенством (15), приведем схему ИНС, реализующей данную модель (рисунок 1).

На рисунке 1 линии (ребра) означают наличие связи, т.е. то, что соответствующий весовой коэффициент отличен от нуля, если ребро не подписано, то весовой коэффициент равен 1, иначе равняется значению подписи. В нейронах указаны используемые функции активации, $x_1$, $x_2$ — входные данные модели.

Формирование модели происходит в 3 этапа:

1 этап. Формирование слоев.

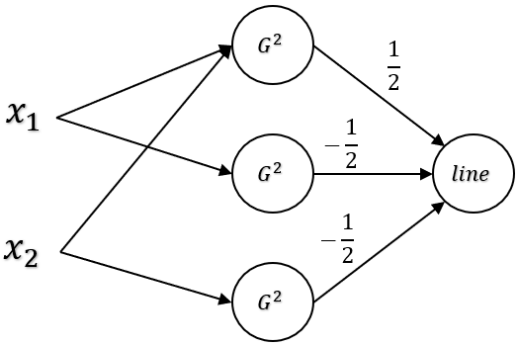



**Рис. 1.** *Схема ИНС, реализующей умножение двух чисел*

2 этап. Создание модели, в том числе ее компиляция.

3 этап. Установка нужных весовых коэффициентов ИНС.

Для тестирования ИНС проведем сравнение точности умножения классическим методом и методом с использованием вышеописанной ИНС. Для этого будем генерировать пары чисел и сравнивать по модулю разность умножения двумя способами. Средняя ошибка на 10000 пар для целых чисел составила 0, для вещественных чисел с 4 знаком после запятой средняя ошибка на 10000 пар составила 0.0317059. При детальном исследовании проблем точности оказывается, что при слишком больших или слишком маленьких результатах вычислений появляются проблемы округления. Т.е. если получаемый результат больше, чем $10^7$ (меньше $10^{-7}$), то ИНС начинает выдавать погрешность вычисления.

**Реализация модели ИНС для градиента потенциального поля**

Покажем теперь непосредственно реализацию модели ИНС для (17), схема ИНС, реализующей градиент потенциального поля для одной координаты, представлена на рисунке 2.

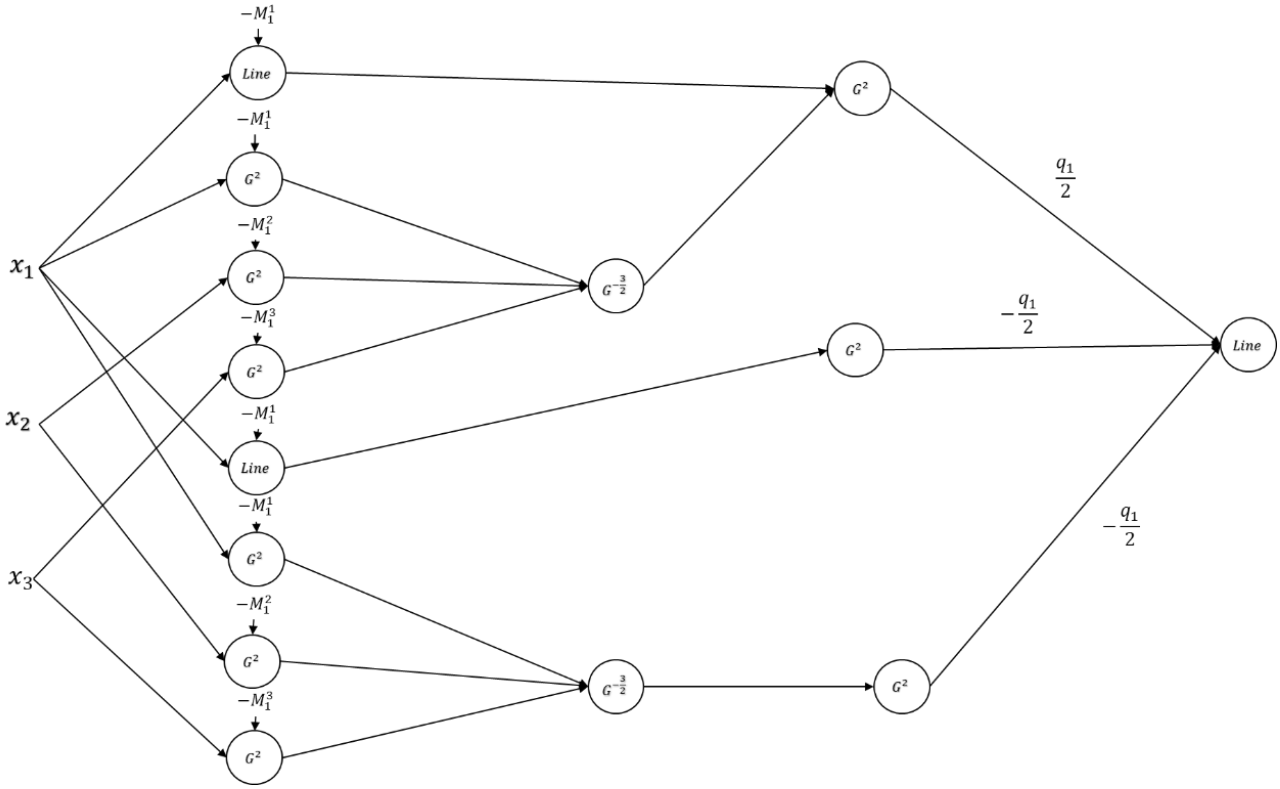



**Рис. 2.** *Схема ИНС, реализующей градиент потенциального поля для одного источника, для* $x_1$ *координаты*

На представленной схеме ИНС обозначения аналогичны рисунку 1. Обозначение в виде стрелки сверху с подписью $-M_1^i$ означает смещение (биас) нейрона, который равняется местоположению

источника. Отсутствие стрелки сверху означает, что смещение данного нейрона равняется 0. Весовые коэффициенты перед выходным слоем, равные $|\frac{q_1}{2}|$, отвечают за мощность источника. На вход ИНС подаются координаты примеси, на выходе ожидается $V_1$. Для получения $V_2$, $V_3$ необходимо незначительно изменить весовые коэффициенты первого слоя, а именно для первого и пятого нейрона первого слоя обнулить веса, отвечающие за первый вход, и установить веса, равные 1, для второго входа для реализации $V_2$. Для $V_3$ необходимо для тех же нейронов обнулить веса, отвечающие за первый вход, и установить веса, равные 1, для третьего входа. Ввиду минимальных отличий в схеме соответствующие схемы для $V_2$, $V_3$ не приводятся.

Приведем результаты тестирования расчета стандартным методом и методом с использованием ИНС, максимальное отклонение расчетов составляет $2.4798^{-07}$, тестирование проводилось для источника мощностью $q = 2$, местоположение источника $M = (0{,}0, -2)$, точки, для которых проводился расчет, были сформированы в виде сетки с шагом 0.1, от $(-1, -1, -1)$ до (1,1,1).

Данная схема производит расчет только для одного источника, для расчета $V_i$ с $K$ источниками необходимо объединить $K$ моделей, описанных на рисунке 2, изменив соответствующим образом весовые коэффициенты, отвечающие за местоположение и мощность источника. После добавить еще один слой, состоящий из одного нейрона, в котором происходит суммирование выходов из каждой модели. Схематично обозначение данной модели представлено на рисунке 3.

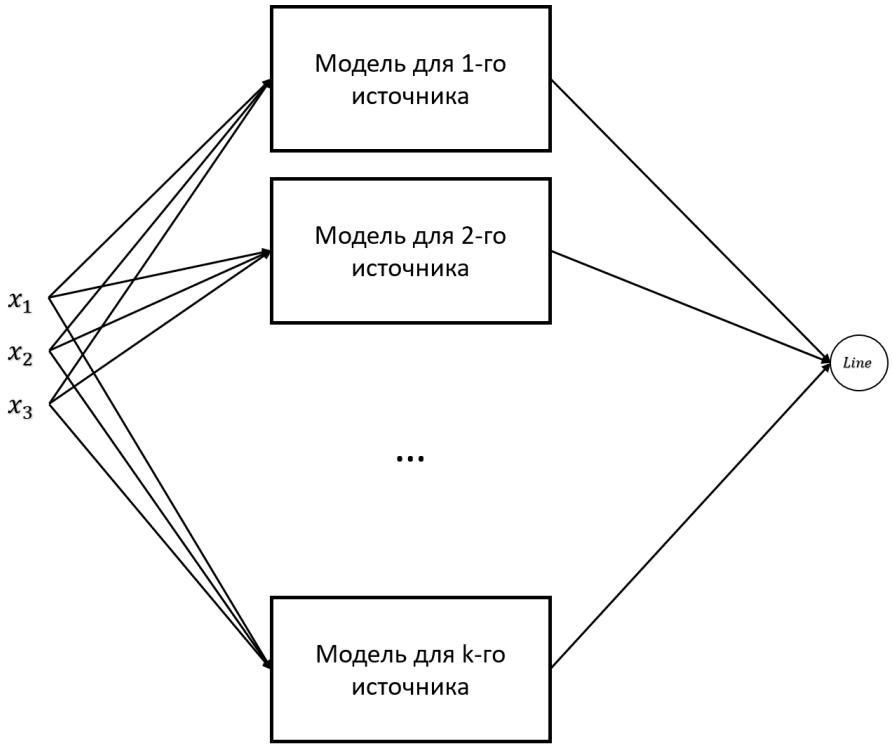



**Рис. 3.** *Схема ИНС для одного $V_i$ с K источниками*

Под каждым квадратом на рисунке 3 подразумевается модель, представленная на рисунке 2. На выходе ожидается значение для определенной координаты $V_i$.

Для реализации схемы ИНС, реализующей все $V_i$ , для каждого $i$ необходимо объединить ИНС, представленные на рисунке 3, в ансамбль моделей и изменить весовые коэффициенты на описанные выше, тогда каждый выход ИНС будет отвечать за свой $V_i$.

Максимальное уклонение расчета градиента потенциального поля, полученного с использованием модели ИНС и без использования ИНС, не превышало $10^{-7}$. Расчетные точки формировались в виде сетки с шагом 0.1, от $(-1, -1, -1)$ до (1,1,1), количество источников варьировалось от 1 до 5, мощность изменялась от 0.01 до 5. Общее количество проведенных тестов с различными конфигурациями местоположений и мощности источников составило 20.

Продемонстрировав возможности реализации (17) в виде модели ИНС, покажем, что данный подход можно использовать для решения задачи Коши.

**Реализация метода Рунге–Кутты в виде модели ИНС**

Покажем, что с использованием ИНС можно реализовать итерационный численный метод решения задачи Коши на примере реализации метода Рунге–Кутты 4-го порядка точности. Строго говоря, итерационный метод лучшим образом представим в виде рекуррентной ИНС, например, в виде ИНС Джордана, описанной в работах [7–9]. Схема ИНС Джордана представлена на рисунке 4 [10].

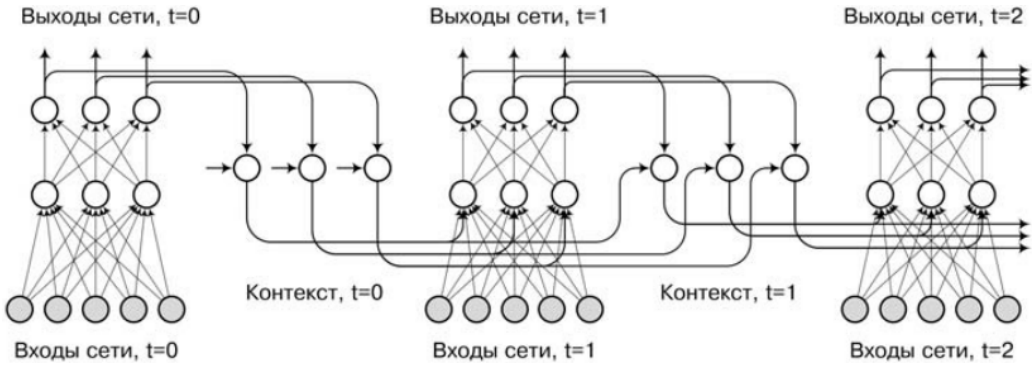



**Рис. 4.** *Рекуррентная нейронная сеть Джордана [10]*

Для применения рекуррентной нейронной сети Джордана для задачи Коши необходимо на вход подавать Н.У. в момент времени $t_0 = 0$, а также шаг по времени h. На выходе сети, подписанном на схеме как «выход сети, $t = 0$», получаем значение примеси в момент времени $t_1 = t_0 + h$. Контекстом в данном случае называется положение примеси в момент времени $t_1$, на вход в данном случае необходимо подавать текущее время и шаг по времени. На выходе будет получено местоположение примеси в момент времени $t_2 = t_1 + h$, после чего процедура повторяется требуемое количество итераций.

Стоит заметить, что современные рекуррентные ИНС с долговременной памятью очень похожи на k-шаговые методы Адамса для численного решения задачи Коши, который часто используют при расчете траекторий движения небесных тел.

Суть данных методов заключается в том, что они используют информацию из k-предыдущих решений. Т.е. для вычисления шага $k + 1$ необходимо знать решения в предыдущих $k$-шагах, в то время как одношаговые методы требуют информации только о предыдущем местоположении точки. Если же говорить в терминах ИНС, то сеть (решатель) должна «помнить» (учитывать) предыдущие k «предсказаний». В ИНС с долговременной памятью используются «вентиль забывания», целью которого является определение, насколько значим шаг по времени, причем для каждого обрабатываемого шага по времени устанавливается свой коэффициент в ходе обучения сети. Данный алгоритм напоминает явный метод Адамса–Башфорта или неявный Адамса–Моултона, которые были представлены в работах [14–16].

Рассмотрим метод Рунге–Кутты 4-го порядка точности и реализуем ИНС, которая проводит вычисления по данному методу. Для этого приведем формулы метода Рунге–Кутты 4-го порядка точности [11]:

$$\vec{y}_{n+1} = \vec{y}_n + \frac{1}{6}h\left(\vec{k}_1^n + 2\vec{k}_2^n + 2\vec{k}_3^n + \vec{k}_4^n\right), \tag{18}$$

$$\vec{k}_1^n = \vec{f}\left(t^n, x_1^n, x_2^n, x_3^n\right), \tag{19}$$

$$\vec{k}_2^n = \vec{f}\left(t^n + \frac{h}{2}, x_1^n + \frac{h}{2}k_1^n, x_2^n + \frac{h}{2}k_1^n, x_3^n + \frac{h}{2}k_1^n\right), \tag{20}$$

$$\vec{k}_3^n = \vec{f}\left(t^n + \frac{h}{2}, x_1^n + \frac{h}{2}k_2^n, x_2^n + \frac{h}{2}k_2^n, x_3^n + \frac{h}{2}k_2^n\right), \tag{21}$$

$$\vec{k}_4^n = \vec{f}\left(t^n + h, x_1^n + hk_3^n, x_2^n + hk_3^n, x_3^n + hk_3^n\right). \tag{22}$$

Если посмотреть на формулу 18, то видно, что там стоит сумма функции правой части, рассчитанная с разными аргументами, т.е. фактически для решения задачи Коши с использованием ИНС необходимо использовать каскад ИНС. Т.е. фактически решение трехмерной задачи Коши представимо в виде суперпозиции функции одной переменной и сложения при условии, что правая часть Д.У. представима в виде суперпозиции функций одной переменной.

Разработанная ИНС «предсказывает» положение примеси на основе полносвязной ИНС. Рекуррентная связь в ИНС реализована путем направления выхода ИНС на вход через цикл, такой подход выбран в целях упрощения реализации ИНС.

Проведем вычислительный эксперимент по моделированию динамики примеси в сталкивающихся потоках. Для этого зададим два источника мощностью $q_1 = q_2 = 1$ и местоположением $M_1(1,0,0)$ и $M_2(-1,0,0)$, в некоторой области около каждого источника будем формировать начальное положение примеси.

Примесью заполнялась цилиндрическая область со следующими параметрами:

1) основание цилиндра по проекции на $yOz$ при $x = 0.45$ и $x = 0.95$ для первого источника и проекция на $yOz$ при $x = -0.45$ и $x = -0.95$ для второго источника;

2) радиус цилиндра $r = 0.125$, ось цилиндра совпадала с осью абсцисс;

3) примесь заполняла сечение цилиндра с шагом 0.01 по оси цилиндра, полученное сечение заполнялось начальными точками по следующим формулам:

$$R(i) = 0.1 + 0.1 \cdot i, \tag{23}$$

$$DF(i) = 8 + 2 \cdot i, \tag{24}$$

$$Y = R(i) \cdot \cos\left((DF(i) \cdot j + (i \cdot j + 1))\,\pi\right)r, \tag{25}$$

$$Z = R(i) \cdot \sin\left((DF(i) \cdot j + (i \cdot j + 1))\,\pi\right)r, \tag{26}$$

$$j = 0,\ldots,9, \qquad i = 0,\ldots,9. \tag{27}$$

Данные формулы использованы для того, чтобы с удалением формирования начальных условий от оси цилиндра увеличивалось количество частиц примеси. Использование генератора случайных чисел для заполнения случайным образом примесью указанных цилиндрических областей вызывает трудности ввиду того, что для моделирования динамики примеси используется различное программное обеспечение, разработанное на разных языках программирования.

Всего начальных условий 22 500 — по 11 250 для каждой из областей, начальное распределение частиц примеси представлено на рисунке 5. Шаг по времени составлял $dt = \frac{1}{2000}$, количество шагов по времени 6000. Моделирование проводилось на вычислительной машине со следующими характеристиками: процессор AMD Ryzen Threadripper 2990WX 32-Core-Processor, 32 ГБ оперативной памяти, с частотой 2 400 МГц и графическим процессором NVIDIA Titan V с 5 120 вычислительными Cuda-ядрами и 12 ГБ видеопамяти.

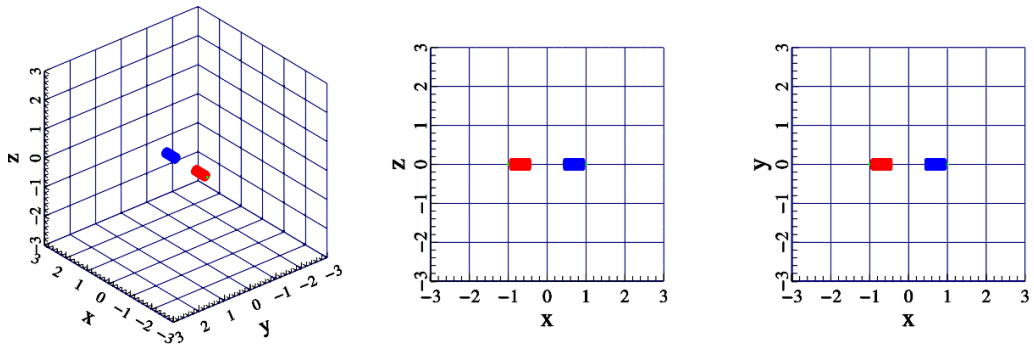



**Рис. 5.** *Начальная конфигурация частиц примеси*

Для моделирования примеси методом Рунге–Кутты 4-го порядка точности применялся программный комплекс, описанный в [2], а также зарегистрированный в ФИПС [17], который проводит параллельные вычисления с использованием технологии OpenMP, а для визуализации — библиотеку mathGL. Кроме того, данный программный комплекс использовался авторами в [1, 12, 13].

Моделирование примеси с использованием описанной выше ИНС происходило в два этапа: на 1 этапе производился расчет динамики примеси с использованием искусственной полносвязной нейронной сети, весовые коэффициенты которой были рассчитаны и установлены на этапе проектирования сети, ИНС реализована на языке программирования Python с использованием библиотеки Keras,

с автоматическим распараллеливанием на видеокарту. После чего полученные данные были записаны в файл и визуализированы с использованием программного комплекса, описанного в [2]. Использование указанного программного комплекса обусловлено тем, что проводилось не только количественное сравнение, но и качественное, а при использовании другой системы визуализации сравнение было бы не таким наглядным. Кроме того, в программном комплексе [2] реализовано распараллеливание визуализации по ядрам процессора, что значительно ускорило процесс визуализации решения.

На рисунках 6–8 представлены изображения для качественного сравнения результатов. Если же говорить о количественном сравнении результатов вычисления, то уклонение метода Рунге–Кутты 4-го порядка точности от метода Рунге–Кутты 4-го порядка точности, реализованного с использованием ИНС, составляет не более $10^{-6}$.

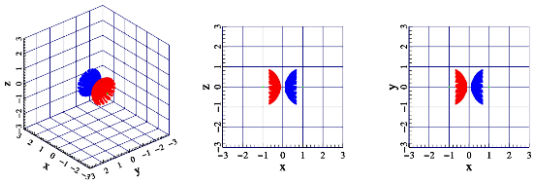
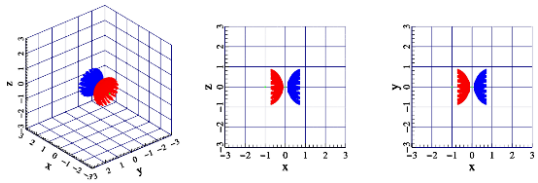
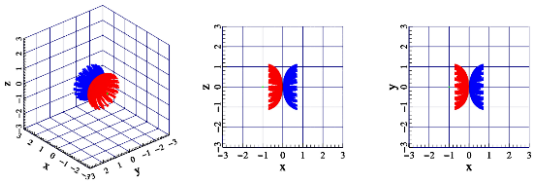
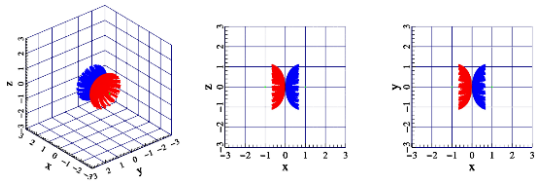
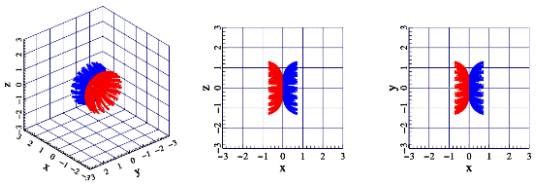
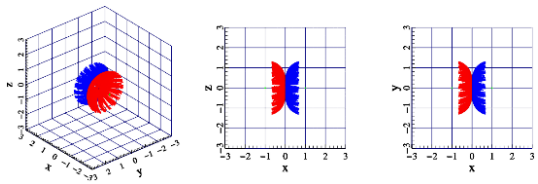
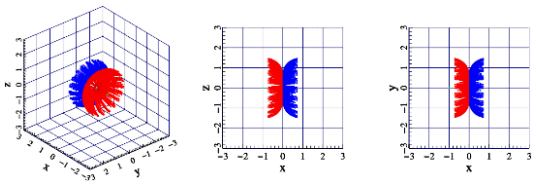
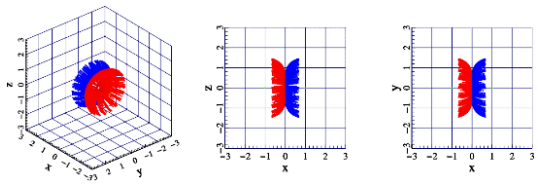
| Время | Расчет динамики примеси с использованием ИНС | Расчет динамики примеси без использования ИНС |
|---|---|---|
| $t = 0.25$ |  |  |
| $t = 0.5$ |  |  |
| $t = 0.75$ |  |  |
| $t = 1.0$ |  |  |

**Рис. 6.** *Сравнение моделирования динамики примеси с использованием ИНС и без,* $0.25 \leqslant t \leqslant 1.0$

**Обсуждение результатов**

Несмотря на то, что скорость моделирования динамики примеси с использованием ИНС ниже по сравнению с программным комплексом [2], возможность применения ИНС для решения задачи Коши имеет большой практический потенциал. Рассмотрим обратную задачу. Такая задача, как было сказано выше, может возникать у экологической полиции для поиска места сброса загрязнений. Пусть есть решения задачи Коши в виде некоторого набора траекторий динамики примеси либо части траекторий в некоторые моменты времени от $t_1$ до некоторого $t_2$, но неизвестны точные характеристики течения, такие как местоположение источника и его мощность. Тогда применение классических алгоритмов обучения ИНС позволит решить данную задачу в ходе обучения, когда на вход ИНС будут поданы местоположения примеси в моменты времени $t_i$, а на выходе из ИНС будут ожидаться местоположения примеси в моменты времени $t_{i+1}$. В ходе обучения необходимо будет подобрать весовые коэффициенты только для 4K-параметров, где K — количество источников, информация о которых будет выбираться исходя из некоторой априорной информации для конкретной задачи.

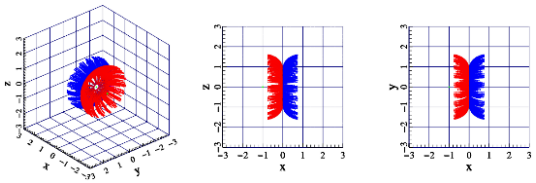
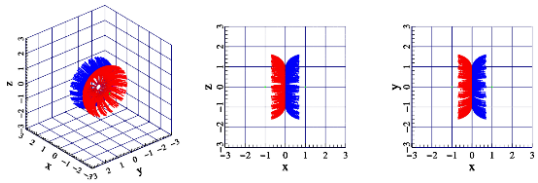
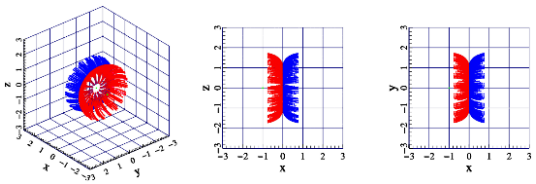
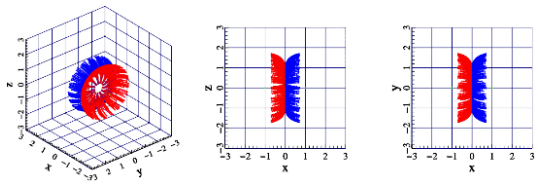
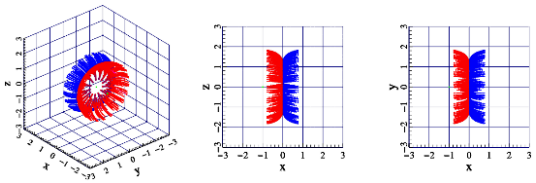
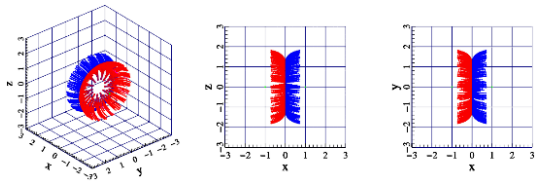
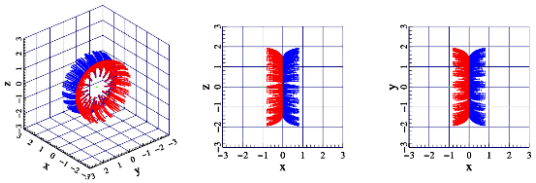
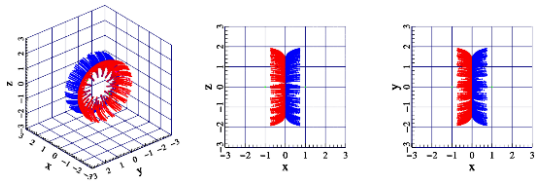
| Время | Расчет динамики примеси с использованием ИНС | Расчет динамики примеси без использования ИНС |
|---|---|---|
| $t = 1.25$ |  |  |
| $t = 1.5$ |  |  |
| $t = 1.75$ |  |  |
| $t = 2.0$ |  |  |

**Рис. 7.** *Сравнение моделирования динамики примеси с использованием ИНС и без,* $1.25 \leqslant t \leqslant 2.0$

Кроме того, использование именно ИНС для решения поставленной задачи позволит решить сразу несколько проблем: избежать необходимости реализации методов градиентного спуска, получить возможность использовать наиболее подходящие модификации данных методов. При использовании библиотек для ИНС с реализованными вычислениями на видеокарте потенциально должно сократиться время на решение обратной задачи поиска конфигурации течения. Кроме того, в дальнейшем можно использовать данный метод для решения сходных задач по поиску конфигураций области течения, например в цилиндрических областях, описанных в работах [12, 13].

На данный момент первые эксперименты показывают возможность применения описанного выше подхода для поиска местоположения источника. Так, были проведены следующие эксперименты по поиску местоположения двух источников с координатами:

$$M'_1\,[0{,}0, -1] \;\text{ и }\; M'_2\,[0{,}0{,}1]\,, \qquad q_1 = q_2 = 1.$$

В качестве обучающих данных использовалась информация о динамике 300 из 22 500 частиц примеси с 1 000 отсчетов по времени, итого обучающих данных было 300 000. В качестве метода обучения использовался стандартный стохастический градиентный спуск без модификаций. Функция потерь (минимизируемая функция) — средняя абсолютная процентная ошибка. Для обучения считалось, что есть априорная информация о том, что сброс примеси происходит с двух источников.

**Обучение ИНС № 1**

Весовые коэффициенты, отвечающие за местоположение источника и его мощность, инициализированные случайными значениями, равны:

$$M_0\colon\ [\,0.04\ \ 0.025\ \ -0.044], \qquad q = -0.01,$$
$$M_1\colon\ [-0.059\ \ -0.0135\ \ 0.034], \qquad q = -0.11.$$

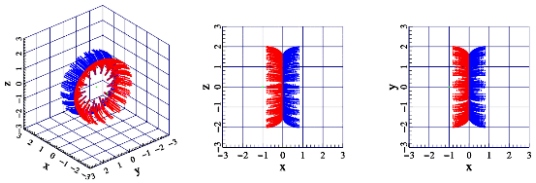
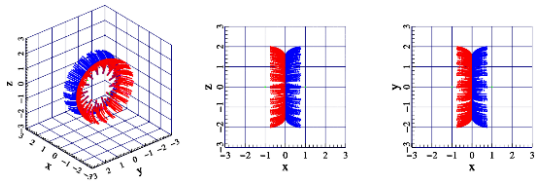
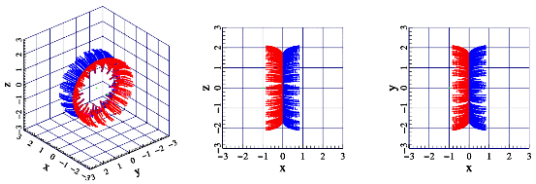
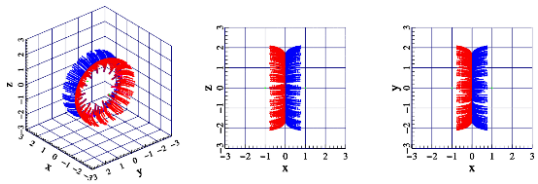
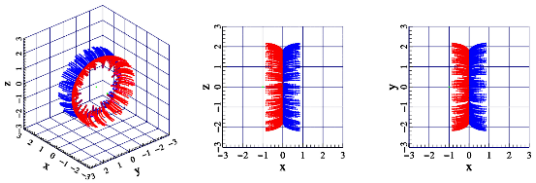
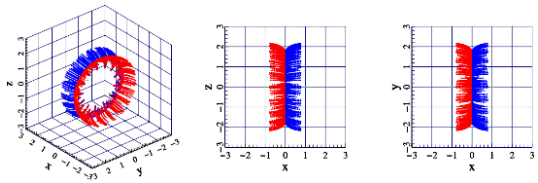
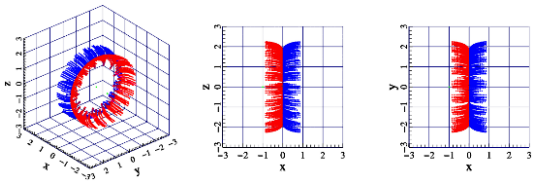
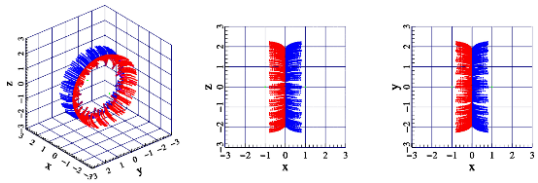
| Время | Расчет динамики примеси с использованием ИНС | Расчет динамики примеси без использования ИНС |
|---|---|---|
| $t = 2.25$ |  |  |
| $t = 2.5$ |  |  |
| $t = 2.75$ |  |  |
| $t = 3.0$ |  |  |

**Рис. 8.** *Сравнение моделирования динамики примеси с использованием ИНС и без,* $2.25 \leqslant t \leqslant 3.0$

Результат поиска, который длился 5 эпох по 3.5 сек каждая.

$$M_0\colon\ [\ 0.021\quad -0.0014\quad -1.0008],\qquad q = 1.009,$$
$$M_1\colon\ [0.0088\quad -0.00284\quad 1.004],\qquad q = 1.0057.$$

В ходе первого обучения получены достаточно хорошие результаты по поиску местоположения. Проведем еще одно обучение, но теперь укажем 1000 эпох с возможностью автоматической остановки обучения при минимальных изменениях потерь (если изменение значения функции потерь будет меньше $10^{-11}$).

**Обучение ИНС № 2**

Весовые коэффициенты, отвечающие за местоположения источников и их мощности, инициализированные случайными значениями, равны:

$$M_0\colon\ [\ 0.0158\ \ 0.0856\ \ -0.0456],\qquad q = -0.006,$$
$$M_1\colon\ [0.0034\ 0.004\ \ 0.041],\qquad q = -0.025.$$

Результат поиска, который длился 75 эпох по 3.5 секунды каждая. Обучение автоматически остановилось ввиду минимальных изменений весов (местоположений и мощностей источника).

$$M_0\colon\ [\ -2.68^{-09}\ \ 3.11^{-09}\ \ -1.00],\qquad q = 1.0,$$
$$M_1\colon\ [-7.66^{-09}\ 2.09^{-08}\ \ 9.99^{-01}],\qquad q = 0.999.$$

Результаты показывают, что в ходе обучения ИНС точно показывает местоположения источников, поэтому проведем еще одно обучение ИНС, в ходе которого у нас будет «ошибочная» информация о количестве источников, а именно будем считать, что идет поиск местоположения 5 источников.

**Обучение ИНС № 3**

$$
\begin{aligned}
&M_0\colon\ [0.0492\ \ 0.045\ \ 0.04], && q = 0.001,\\
&M_1\colon\ [0.14\ \ \ 0.022\ \ -0.01], && q = -0.058,\\
&M_2\colon\ [0.105\ \ -0.003\ \ -0.0298], && q = 0.003,\\
&M_3\colon\ [-0.019\ \ -0.063\ \ \ -0.049], && q = 0.007,\\
&M_4\colon\ [0.017\ \ \ 0.027\ \ 0.119], && q = -0.0047.
\end{aligned}
$$

Результат поиска:

$$
\begin{aligned}
&M_0\colon\ [-0.023\ \ 0.006\ \ 0.165], && q = 1.15^{-06},\\
&M_1\colon\ [-0.005\ \ \ 0.012\ \ \ -0.213], && q = -3.49^{-06},\\
&M_2\colon\ [5.8^{-05}\ \ -5.51^{-05}\ \ -9.91^{-01}], && q = 0.471,\\
&M_3\colon\ [-5.54^{-05}\ \ 5.31^{-05}\ \ \ -1.01], && q = 0.529,\\
&M_4\colon\ [3.35^{-07}\ \ \ -9.36^{-08}\ \ 1.00], && q = 1.00.
\end{aligned}
$$

Обучение длилось на протяжении 90 эпох по 4.5 секунды и автоматически остановилось ввиду минимальных изменений весов (местоположений и мощностей источников).

По результатам обучения ИНС показала корректные местоположения источников, т.к. источники $M_0$ и $M_1$ имеют мощность, близкую к 0, а следовательно, можно ими пренебречь. Источники $M_2$ и $M_3$ имеют местоположения, близкие к искомому местоположению $M_1'$, и в сумме их мощность равняется 1. А источник $M_4$ с достаточно хорошей точностью близок к источнику $M_2'$ как по местоположению, так и по мощности.

Видно, что использование метода обучения ИНС для поиска конфигурации течения имеет смысл, но необходимо дальнейшее исследование для поиска условий применения, степени значимости априорной информации о течении, а также поиска необходимого количества начальных данных, достаточных для решения задачи.

## ЛИТЕРАТУРА

1. Галкин В. А., Смородинов А. Д., Моргун Д. А. Решение уравнения Навье–Стокса для сталкивающихся потоков. *Успехи кибернетики.* 2023;4(2):8–15. DOI: 10.51790/2712-9942-2023-4-2-01.
2. Смородинов А. Д., Гавриленко Т. В., Дубовик А. О., Моргун Д. А. Реализация программного обеспечения для решения задачи Коши с различными конфигурациями полей начальных данных и визуализации результатов математического моделирования течения тяжелой примеси. *Успехи кибернетики.* 2024;5(2):35–45. DOI: 10.51790/2712-9942-2024-5-2-04.
3. Колмогоров А. Н. О представлении непрерывных функций нескольких переменных в виде суперпозиций непрерывных функций одного переменного и сложения. *Докл. АН СССР.* 1957;114(5):953–956.
4. Weierstrass K. Uber die analystische Darstellbarkeit sogenannter willkűrlicher Funktionen reeller Argumente. *Sitzungsbericht der Kőnig. Academie der Wissenschaften*. 1885.
5. Sprecher D. On the Structure of Continuous Functions of Several Variables. *Transactions of the American Mathematical Society*. 1965;115(3):340–355. DOI: 10.2307/1994273.
6. Cybenko G. Approximation by Superpositions of a Sigmoidal Function. *Mathematics of Control, Signals, and Systems.* 1989;2(4):303–314. CiteSeerX: 10.1.1.441.7873. DOI: 10.1007/BF02551274.
7. Jordan M. I. *Serial Order: A Parallel Distributed Processing Approach*. ICS Report 8604.Institute for Cognitive Science, University of California, San Diego, 1986.
8. Jordan M. I. Attractor Dynamics and Parallelism in a Connectionist Sequential Machine. *Proceedings of the Annual Meeting of the Cognitive Science Society*. 1986:8
9. Jordan M. I. Serial Order: A Parallel Distributed Processing Approach. *Advances in Psychology*. 1997;121:471–495.
10. Николенко С., Кадурин А., Архангельская Е. *Глубокое обучение*. СПб.: Питер, 2018. 480 с. ISBN: 978-5-496-02536-2.

11. Калиткин Н. Н. *Численные методы.* СПб.: БХВ-Петербург; 2011. 592 с.
12. Галкин В. А., Дубовик А. О., Смородинов А. Д. Моделирование и визуализация седиментации в течении вязкой несжимаемой жидкости. *Научная визуализация*. 2024;16(3):71–78. DOI: 10.26583/sv.16.3.07. EDN: KNUTBP.
13. Галкин В. А., Дубовик А. О., Смородинов А. Д. Моделирование течения вязкой несжимаемой жидкости с учетом наличия примеси в поле течения. *Информатика, управляющие системы, математическое и компьютерное моделирование (ИУСМКМ-2024).* XV Международная научно-техническая конференция в рамках X Международного научного форума Донецкой Народной Республики. Донецк: Донецкий национальный технический университет; 2024:132–139. EDN: WNJHRZ.
14. Butcher J. C. *Numerical Methods for Ordinary Differential Equations*. 2 edition. New Zealand: Wiley, 2003. ISBN: 0-471-96758-0.
15. Hairer E., Nørsett S. P., Wanner G. *Solving Ordinary Differential Equations I: Nonstiff Problems.* 2 edition. Berlin: Springer-Verlag; 2008. ISBN: 978-3-540-56670-0.
16. Cash J. R., Karp Alan H. A Variable Order Runge–Kutta Method for Initial Value Problems with Rapidly Varying Right-Hand Sides. *ACM Trans. Math. Softw*. 1990;16(3):201–222.
17. *Программный комплекс для решения задачи Коши с различными конфигурациями полей начальных данных и визуализации результатов математического моделирования течения тяжелой примеси*: свидетельство о государственной регистрации программы для ЭВМ № 2024689709, Российская Федерация / Смородинов А. Д., Гавриленко Т. В., Дубовик А. О., Моргун Д. А. EDN: XBPJPY.