

Министерство образования и науки Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего профессионального образования
«Московский государственный технический университет имени Н. Э. Баумана»

Кафедра «Прикладная математика»



Курсовая работа

по дисциплине «Методы вычислений»

Решение задачи Коши для ОДУ первого порядка с помощью искусственных нейронных сетей

Выполнил студент группы ФН2-62

Никуленков М.Р.

*Научный
руководитель* ассистент кафедры ФН-2

Лазарев А.А.

Оглавление

1. Введение	3
2. Формулировка задач курсовой работы	6
3. Описание метода решения	7
4. Анализ результатов	9
4.1. Тест 1	9
4.2. Тест 2	12
4.3. Тест 3	14
4.4. Общий анализ	16
5. Заключение	18
Список литературы	19

1. Введение

На сегодняшний день разработано множество методов решения дифференциальных уравнений. Некоторые из них дают ответ в виде решения в выбранной группе точек, другие используют базисные функции для представления решения. Другой подход к решению дифференциальных уравнений основан на возможности аппроксимировать функции с помощью нейронных сетей прямого распространения.

Искусственная нейронная сеть (ИНС) – это математическая модель, основной частью которой является искусственный нейрон, осуществляющий нелинейное преобразование суммы произведения входных сигналов на весовые коэффициенты. Она построена по образу и подобию сетей нервных клеток живого организма. ИНС представляет собой систему соединённых и взаимодействующих между собой искусственных нейронов. Каждый из нейронов подобной сети имеет дело только с сигналами, которые он получает, и сигналами, которые он посылает другим нейронам.

Персептроном называют ИНС с одним нейроном. Рассмотрим математическую модель персептрона:

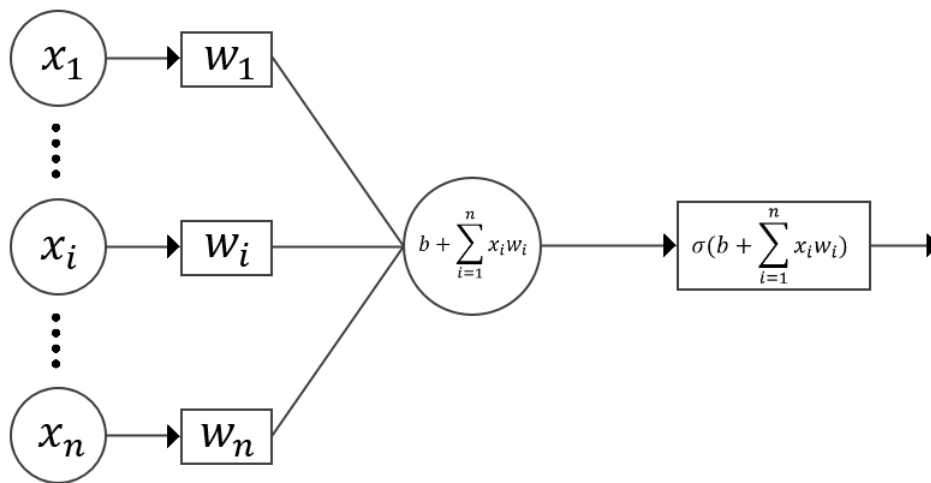


Рис. 1.

На рис. 1 изображена структура персептрона. x_1, \dots, x_n – входные данные, w_1, \dots, w_n – весовые коэффициенты, b – сдвиг и $\sigma(x)$ – активирующая функция.

Работу персептрона можно разбить на следующие шаги:

1. Входные данные умножаются на соответствующие им весовые коэффициенты

и суммируются: $\sum_{i=1}^n x_i w_i$.

2. К сумме прибавляется смещение: $b + \sum_{i=1}^n x_i w_i$.

3. Считается результирующее значение персептрона: $\sigma \left(b + \sum_{i=1}^n x_i w_i \right)$.

Рассмотрим структуру нейронной сети прямого распространения:

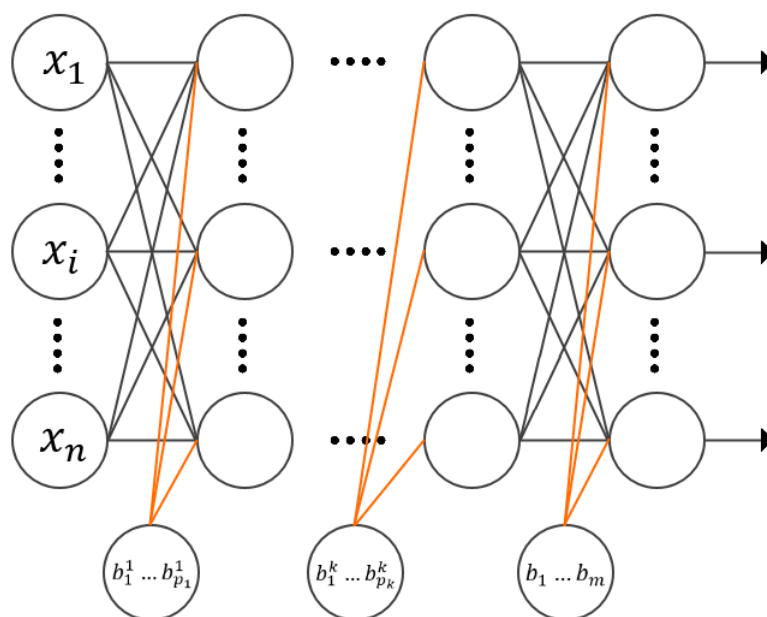


Рис. 2.

На рис. 2 изображена структура ИНС прямого распространения с k скрытыми слоями и m результирующими нейронами. x_1, \dots, x_n – входные данные, b_j – смещение нейронов на j -м слое, b – смещение нейронов на результирующем слое. Каждому нейрону в j -м слое соответствует p_{j-1} весовых коэффициентов.

Нейроны в сетях такого типа делятся на группы с общим выходным сигналом – слои, при этом на каждый нейрон первого слоя подаются все элементы внешнего выходного сигнала, а все выходы нейронов слоя подаются на каждый нейрон следующего слоя.

Нейроны выполняют взвешенное суммирование элементов выходных сигналов, затем к данной сумме прибавляется смещение нейрона. Над полученным результатом затем выполняется преобразование с помощью активационной функции. Значение функции активации есть выход нейрона.

Решение ОДУ на сетке может быть представлено в виде ИНС прямого распространения с одним результирующим нейроном. В качестве входных данных в данном случае выступает множество точек сетки. В работе рассматривается ИНС прямого распространения с одним скрытым слоем.

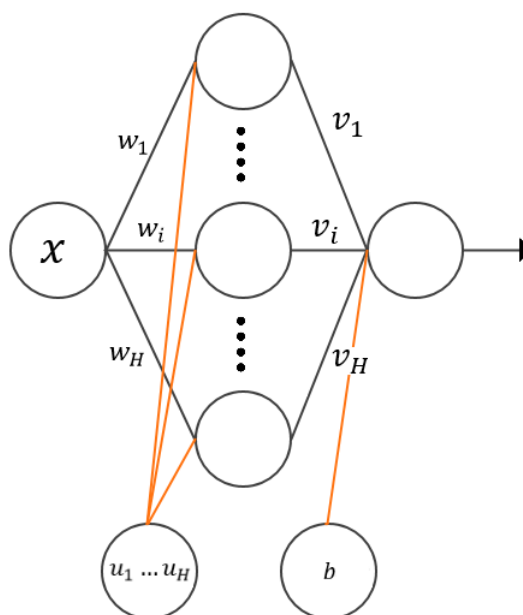


Рис. 3. ИНС прямого распространения с одним скрытым слоем.

Необходимо обучить нейронную сеть таким образом, чтобы она удовлетворяла решению дифференциального уравнения. Для решения поставленной задачи требуется функционал, описывающий качество нейронной сети. Обычно его называют «функцией потерь» и обозначают $L[\vec{p}]$, где \vec{p} – параметры сети. Функция потерь характеризует величину отклонения полученного ответа от правильного ответа.

Данный подход к решению дифференциальных уравнений основан на корректировании весовых коэффициентов и сдвигов нейронов таким образом, чтобы минимизировать функцию потерь.

2. Формулировка задач курсовой работы

Реализовать алгоритм решения задачи Коши для ОДУ первого порядка с помощью искусственных нейронных сетей. Продемонстрировать работу алгоритма, сравнить результат с численными методами решения других классов. Выделить достоинства и недостатки метода.

3. Описание метода решения

Рассмотрим задачу Коши для ОДУ первого порядка:

$$\frac{dy}{dx} = f(x, \psi), \psi(0) = A, x \in [0, 1].$$

Представим решение в форме, которая соответствует начальным условиям независимо от нейронной сети. Решение может быть представлено в виде $\psi_t(x, \vec{p}) = A + xN(x, \vec{p})$, где $N(x, \vec{p})$ – нейронная сеть с одним результирующим нейроном, параметры \vec{p} которой должны быть определены таким образом, чтобы $\psi_t(x, \vec{p})$ аппроксимировала решение. Таким образом, поиск решения в данной форме преобразует изначальную задачу ограниченной оптимизации к более простой задаче неограниченной оптимизации.

Необходимо найти решение на сетке x_1, x_2, \dots, x_K . Функция потерь определяется как:

$$L[\vec{p}] = \sum_{i=1}^K \left\{ \frac{d\psi_t(x_i)}{dx} - f(x_i, \psi_t(x_i)) \right\}^2.$$

Для минимизации $L[\vec{p}]$ в работе применяется метод градиентного спуска с дроблением шага.

На рис. 3 изображена нейронная сеть с одним скрытым слоем, содержащим H нейронов, где $w_1 \dots w_H$ – весовые коэффициенты входного значения, $u_1 \dots u_H$ – смещения нейронов скрытого слоя, b – смещение результирующего нейрона. В качестве активирующей функции нейронов в скрытом слое используется сигмоида $\sigma(x) = \frac{1}{1 + e^{-x}}$, для результирующего нейрона используется линейная функция $f(x) = x$.

Основная сложность данного подхода к решению ОДУ заключается в расчете производных. Формулы для расчета градиента функции потерь относительно параметров:

$$L[\vec{p}] = \sum_{i=1}^K \left\{ \frac{d\psi_t(x_i)}{dx} - f(x_i, \psi_t(x_i, \vec{p})) \right\}^2 = \sum_{i=1}^K \left\{ N(x_i, \vec{p}) + x_i \frac{\partial N(x, \vec{p})}{\partial x} - f(x_i, \psi_t(x_i, \vec{p})) \right\}^2$$

$$\begin{aligned} \frac{\partial L[\vec{p}]}{\partial w_j} &= 2 \sum_{i=1}^K \left(N(x_i, \vec{p}) + x_i \frac{\partial N(x_i, \vec{p})}{\partial x} - f(x_i, \psi_t(x_i, \vec{p})) \right) \cdot \\ &\cdot \left(\frac{\partial N(x_i, \vec{p})}{\partial w_j} + x_i \frac{\partial N(x_i, \vec{p})}{\partial x \partial w_j} - \frac{\partial f(x_i, \psi_t(x_i, \vec{p}))}{\partial \psi_t(x_i, \vec{p})} \cdot x_i \cdot \frac{\partial N(x_i, \vec{p})}{\partial w_j} \right) \\ \frac{\partial L[\vec{p}]}{\partial u_j} &= 2 \sum_{i=1}^K \left(N(x_i, \vec{p}) + x_i \frac{\partial N(x_i, \vec{p})}{\partial x} - f(x_i, \psi_t(x_i, \vec{p})) \right) \cdot \\ &\cdot \left(\frac{\partial N(x_i, \vec{p})}{\partial u_j} + x_i \frac{\partial N(x_i, \vec{p})}{\partial x \partial u_j} - \frac{\partial f(x_i, \psi_t(x_i, \vec{p}))}{\partial \psi_t(x_i, \vec{p})} \cdot x_i \cdot \frac{\partial N(x_i, \vec{p})}{\partial u_j} \right) \end{aligned}$$

$$\begin{aligned}
\frac{\partial L[\vec{p}]}{\partial v_j} &= 2 \sum_{i=1}^K \left(N(x_i, \vec{p}) + x_i \frac{\partial N(x_i, \vec{p})}{\partial x} - f(x_i, \psi_t(x_i, \vec{p})) \right) \cdot \\
&\cdot \left(\frac{\partial N(x_i, \vec{p})}{\partial v_j} + x_i \frac{\partial N(x_i, \vec{p})}{\partial x \partial v_j} - \frac{\partial f(x_i, \psi_t(x_i, \vec{p}))}{\partial \psi_t(x_i, \vec{p})} \cdot x_i \cdot \frac{\partial N(x_i, \vec{p})}{\partial v_j} \right) \\
\frac{\partial L[\vec{p}]}{\partial b} &= 2 \sum_{i=1}^K \left(N(x_i, \vec{p}) + x_i \frac{\partial N(x_i, \vec{p})}{\partial x} - f(x_i, \psi_t(x_i, \vec{p})) \right) \cdot \\
&\cdot \left(1 - \frac{\partial f(x_i, \psi_t(x_i, \vec{p}))}{\partial \psi_t(x_i, \vec{p})} \cdot x_i \right)
\end{aligned}$$

Обозначим $z_j = (w_j x + u_j)$.

Свойство сигмoиды: $\sigma'(x) = \sigma(x)(1 - \sigma(x))$.

$$\begin{aligned}
N(x, \vec{p}) &= b + \sum_{j=1}^H v_j \sigma(z_j) \\
\frac{\partial N(x, \vec{p})}{\partial x} &= \sum_{i=1}^H v_i \cdot w_i \cdot \sigma'(z_i) \\
\frac{\partial N(x, \vec{p})}{\partial w_j} &= x \cdot v_j \cdot \sigma'(z_j) \\
\frac{\partial N(x, \vec{p})}{\partial u_j} &= v_j \cdot \sigma'(z_j) \\
\frac{\partial N(x, \vec{p})}{\partial v_j} &= x \cdot \sigma(z_j) \\
\frac{\partial N(x, \vec{p})}{\partial x \partial w_j} &= x \cdot \sigma''(z_j) \cdot w_j \cdot v_j + \sigma'(z_j) \cdot v_j \\
\frac{\partial N(x, \vec{p})}{\partial x \partial v_j} &= \sigma'(z_j) \cdot w_j \\
\frac{\partial N(x, \vec{p})}{\partial x \partial u_j} &= \sigma''(z_j) \cdot w_j \cdot v_j.
\end{aligned}$$

4. Анализ результатов

Рассмотрим зависимость максимальной относительной погрешности и значения функции потерь от количества нейронов в скрытом слое. Для нейронной сети с 4-мя нейронами в скрытом слое рассмотрим решения нескольких дифференциальных уравнений при различных начальных значениях сдвига результирующего нейрона b_{init} : возьмем нулевое значение, а так же экспериментально выбранное значение, соответствующее меньшему значению функции потерь. В рассматриваемой модели нейронной сети начальные значения остальных параметров принимаются за 0. Сравним результаты с решениями, полученными методами Эйлера и Рунге-Кутты 4-го порядка. Покажем зависимость относительной погрешности от количества проведенных итераций (минимизация прекращалась, когда норма градиента становилась меньше 10^{-4} , либо при заиклиивании алгоритма). Для каждого теста на сетке с шагом 0.1 покажем, что выбор алгоритма минимизации позволяет улучшить результат: будем проводить градиентный спуск с фиксированным шагом, равным 10^{-10} .

4.1. Тест 1

$$\frac{du}{dx} = -2u, u(0) = 2.$$

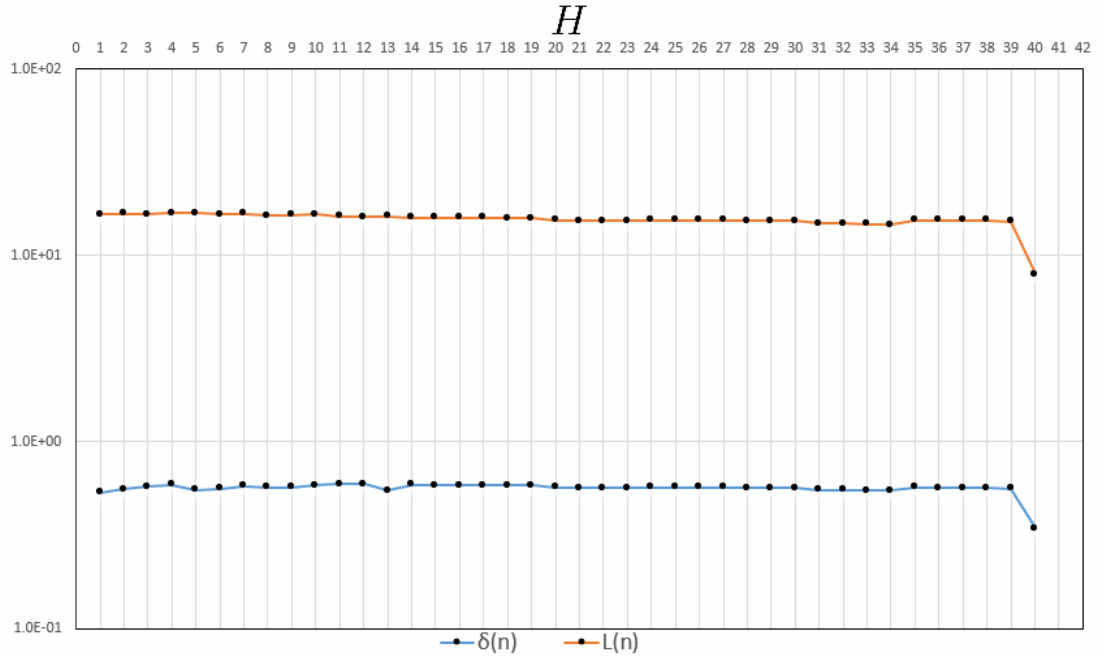


Рис. 4. Логарифмические графики зависимости макс. отн. погр-ти и значения ф-ии потерь от кол-ва нейронов, $b_{init} = 0$.

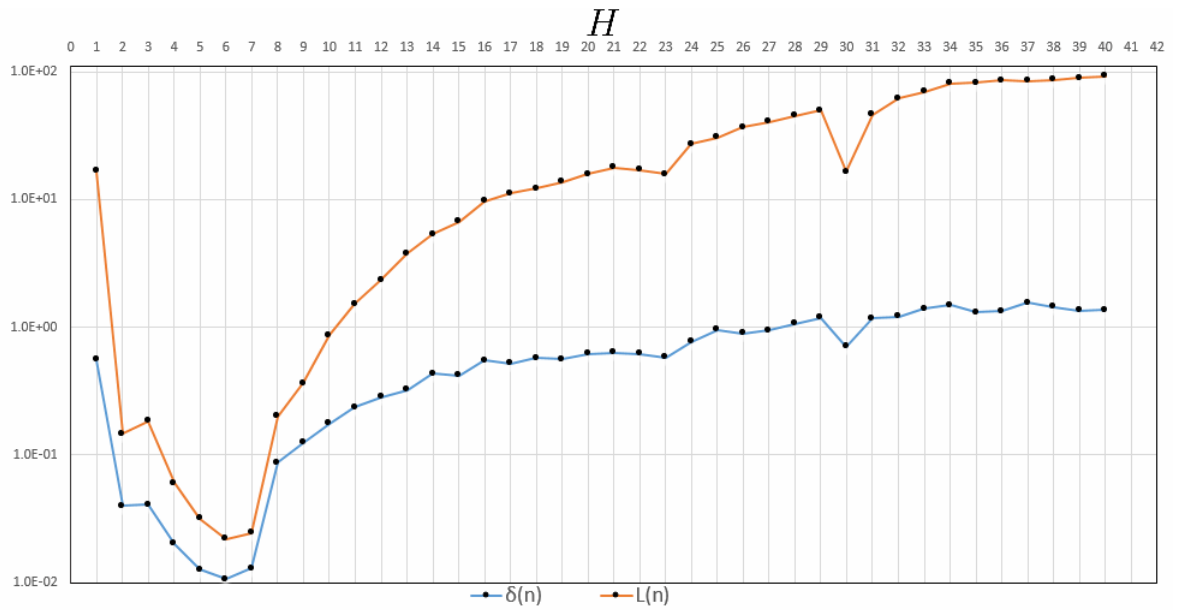


Рис. 5. Логарифмические графики зависимости макс. отн. погр-ти и значения ф-ии потерь от кол-ва нейронов, $b_{init} = -1.49$.

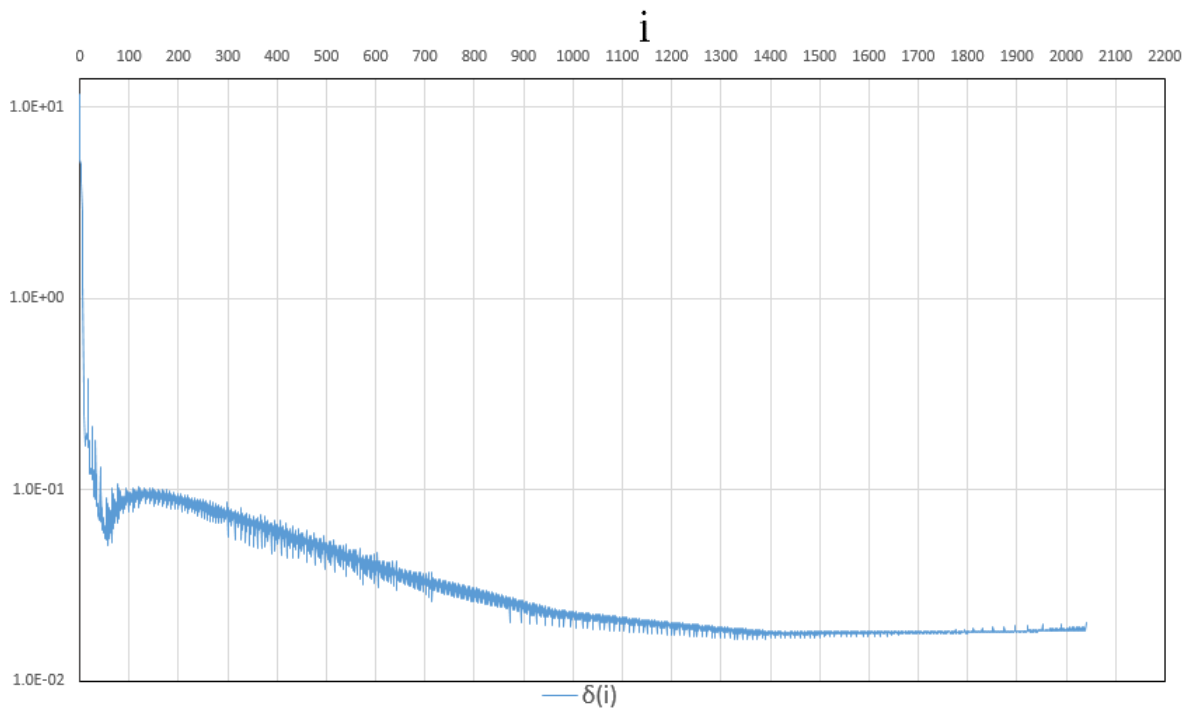


Рис. 6. Логарифмический график зависимости макс. отн. погр-ти от кол-ва итераций, $b_{init} = -1.49$, $H = 4$.

Сравнение нейронной сети с 4-мя нейронами в скрытом слое, методов Эйлера и Рунге-Кутты 4-го порядка:

Метод	Шаг	Абс. погр.	Отн. погр.	Порядок	Время (нс)
Нейр. сеть, $b = 0$	0.2	0.40951	0.58761	-	16.87
	0.1	0.40951	0.58761		26.98
	0.05	0.40951	0.59246		56.57
	0.025	0.40953	0.59246		110.58
Нейр. сеть, $b = -1.49$	0.2	0.01306	0.02054	-	730.61
	0.1	0.01306	0.02022		2171.94
	0.05	0.01367	0.02022		3971.92
	0.025	0.01367	0.02059		9143.66
Эйлера	0.2	0.17866	0, 42542	1	0.02
	0.1	0.08040	0.20661		0.04
	0.05	0.03840	0.10166		0.04
	0.025	0.01879	0.05042		0.08
Рунге - Кутты	0.2	2.16E-04	3.59E-04	4	0.02
	0.1	1.16E-05	2.32E-05		0.04
	0.05	6.66E-07	1.33E-06		0.06
	0.025	4.01E-08	7.99E-08		0.11

Расчеты при $b_{init} = -1.49$ с фиксированным шагом, равным 10^{-10} , дали ответ с относительной погрешностью 0.01123 за 1643 секунды.

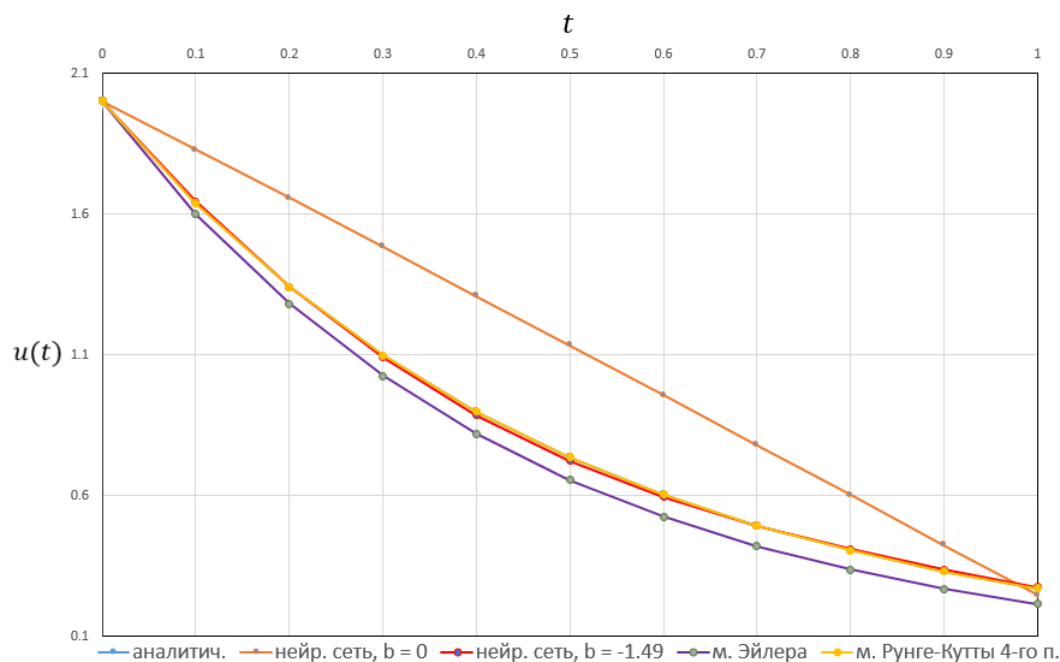


Рис. 7. Графики полученных решений.

4.2. Тест 2

$$\frac{du}{dx} = u^2 x, u(0) = 1.$$

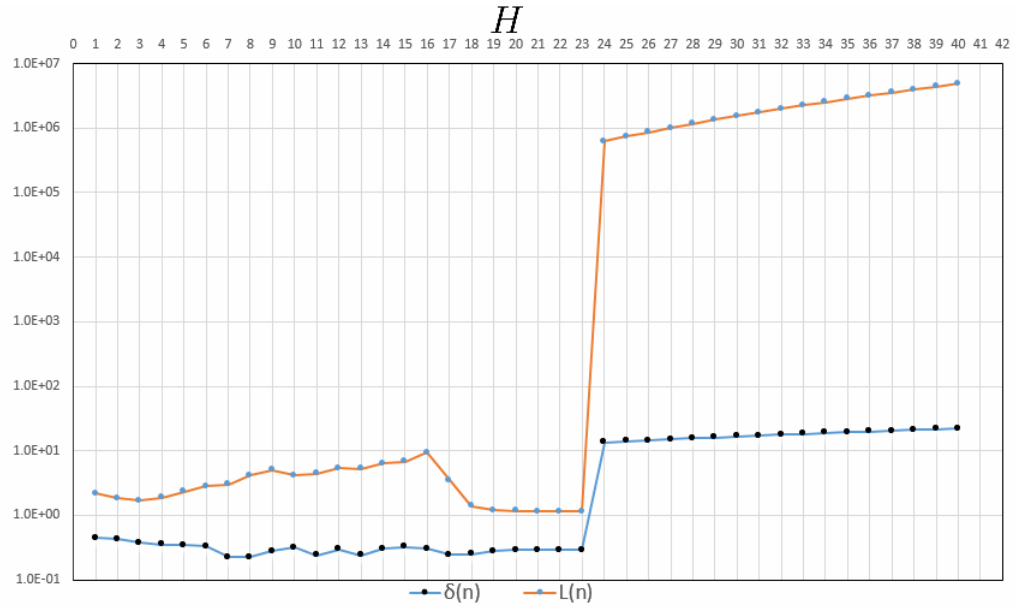


Рис. 8. Логарифмические графики зависимости макс. отн. погр-ти и значения ф-ии потерь от кол-ва нейронов, $b_{init} = 0$.

Эксперименты показали, что в данном тесте нулевое начальное значение сдвига результирующего нейрона соответствует меньшему значению функции потерь. Другие значения не рассматриваются.

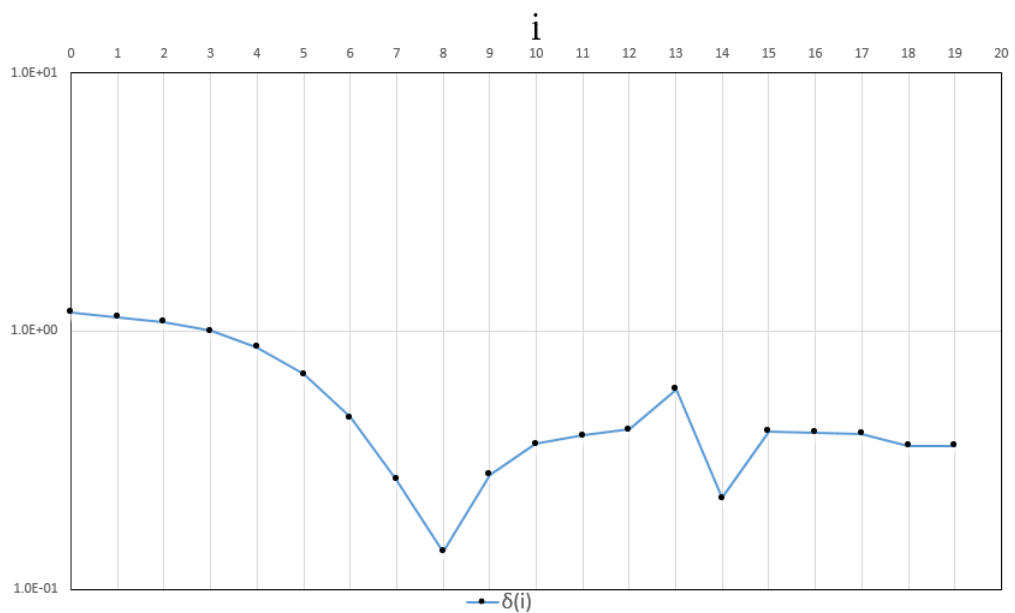


Рис. 9. Логарифмический график зависимости макс. отн. погр-ти от кол-ва итераций, $b_{init} = 0$, $H = 4$.

Сравнение нейронной сети с 4-мя нейронами в скрытом слое, методов Эйлера и Рунге-Кутты 4-го порядка:

Метод	Шаг	Абс. погр.	Отн. погр.	Порядок	Время (нс)
Нейр. сеть, $b = 0$	0.2	0.69918	0.34959	-	17.27
	0.1	0.69918	0.34959		25.01
	0.05	0.69918	0.34959		44.59
	0.025	0.69918	0.34959		64.14
Эйлера	0.2	0.45952	0.22976	1	0.03
	0.1	0.28715	0.14357		0.03
	0.05	0.16577	0.08289		0.03
	0.025	0.09019	0.04510		0.09
Рунге - Кутты	0.2	2.04E-05	2.14E-05	4	0.03
	0.1	1.24E-06	1.26E-06		0.05
	0.05	6.71E-08	6.61E-08		0.06
	0.025	4.02E-09	4.03E-09		0.12

Расчеты при $b_{init} = 0$ с фиксированным шагом, равным 10^{-10} , дали ответ с относительной погрешностью 0.20659 за 936 секунд.

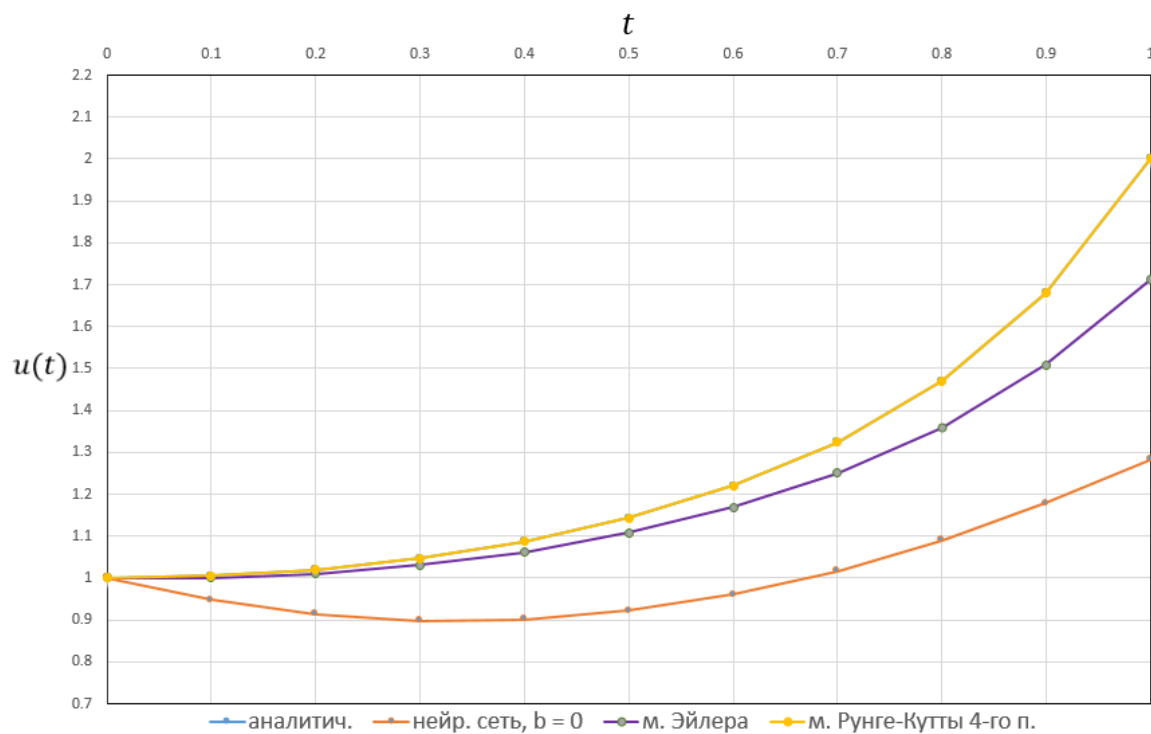


Рис. 10. Графики полученных решений.

4.3. Тест 3

$$\frac{du}{dx} = x^3 + 2x + x^2 \frac{(1 + 3x^2)}{1 + x + x^3} - u \left(x + \frac{(1 + 3x^2)}{1 + x + x^3} \right), u(0) = 1.$$

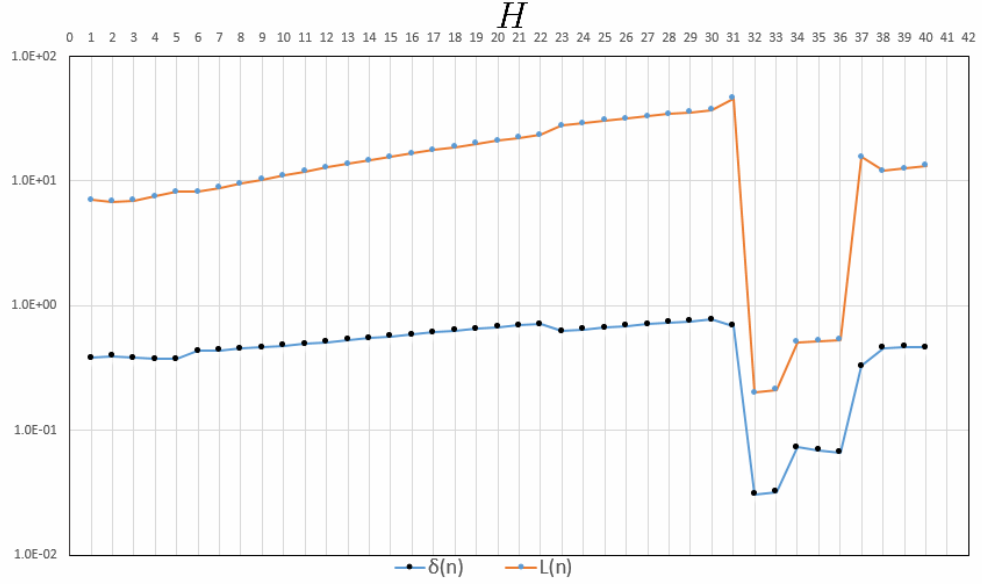


Рис. 11. Логарифмические графики зависимости макс. отн. погр-ти и значения ф-ии потерь от кол-ва нейронов, $b_{init} = 0$.

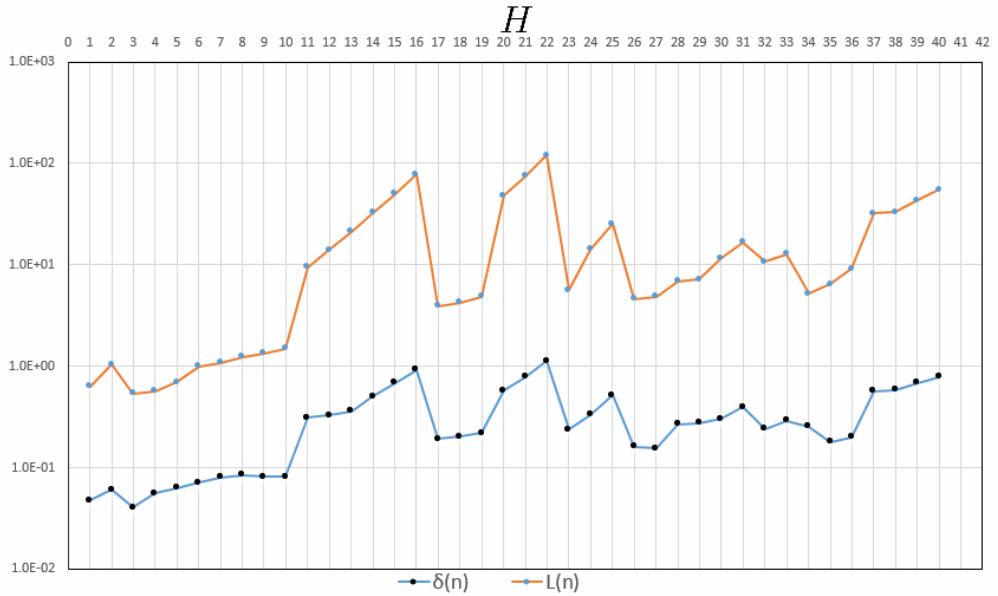


Рис. 12. Логарифмические графики зависимости макс. отн. погр-ти и значения ф-ии потерь от кол-ва нейронов, $b_{init} = -2$.

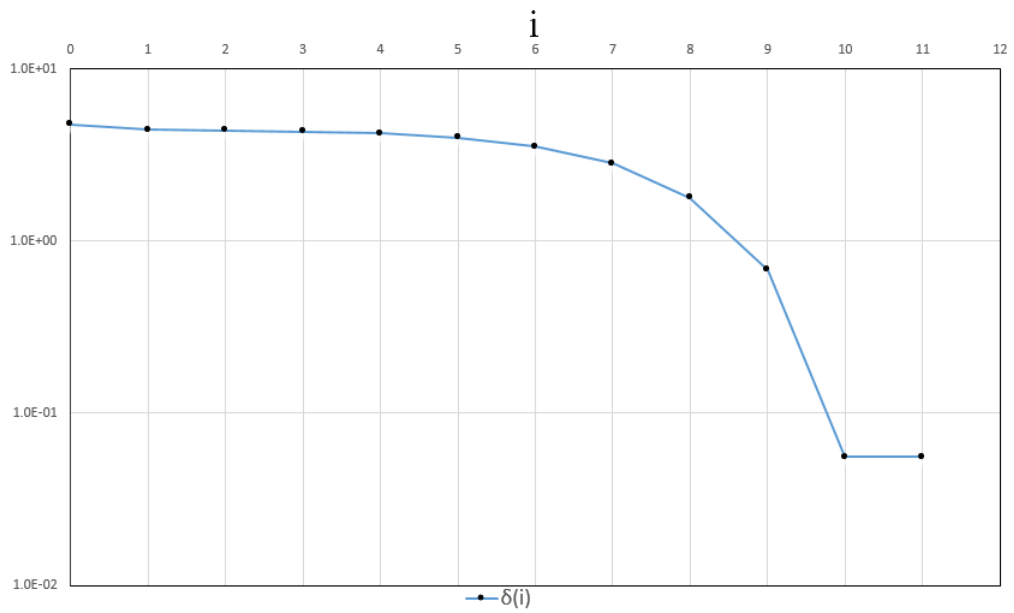


Рис. 13. Логарифмический график зависимости макс. отн. погр-ти от кол-ва итераций, $b_{init} = -2$, $H = 4$.

Сравнение нейронной сети с 4-мя нейронами в скрытом слое, методов Эйлера и Рунге-Кутты 4-го порядка:

Метод	Шаг	Абс. погр.	Отн. погр.	Порядок	Время (нс)
Нейр. сеть, $b = 0$	0.2	0.27362	0.33370	-	10.74
	0.1	0.27734	0.34971		18.76
	0.05	0.27816	0.34971		34.38
	0.025	0.27841	0.34971		64.04
Нейр. сеть, $b = -2$	0.2	0.04599	0.05401	-	10.56
	0.1	0.04599	0.05583		19.82
	0.05	0.04740	0.05722		33.09
	0.025	0.04740	0.05722		67.48
Эйлера	0.2	0.13038	0.12971	1	0.03
	0.1	0.06167	0.06246		0.03
	0.05	0.03004	0.03031		0.04
	0.025	0.01483	0.01493		0.1
Рунге - Кутты	0.2	2.31E-04	2.30E-04	4	0.03
	0.1	1.30E-05	1.30E-05		0.05
	0.05	7.63E-07	7.64E-07		0.08
	0.025	4.63E-08	4.63E-08		0.15

Расчеты при $b_{init} = -2$ с фиксированным шагом, равным 10^{-10} , дали ответ с относительной погрешностью 0.14362 за 1016 секунд.

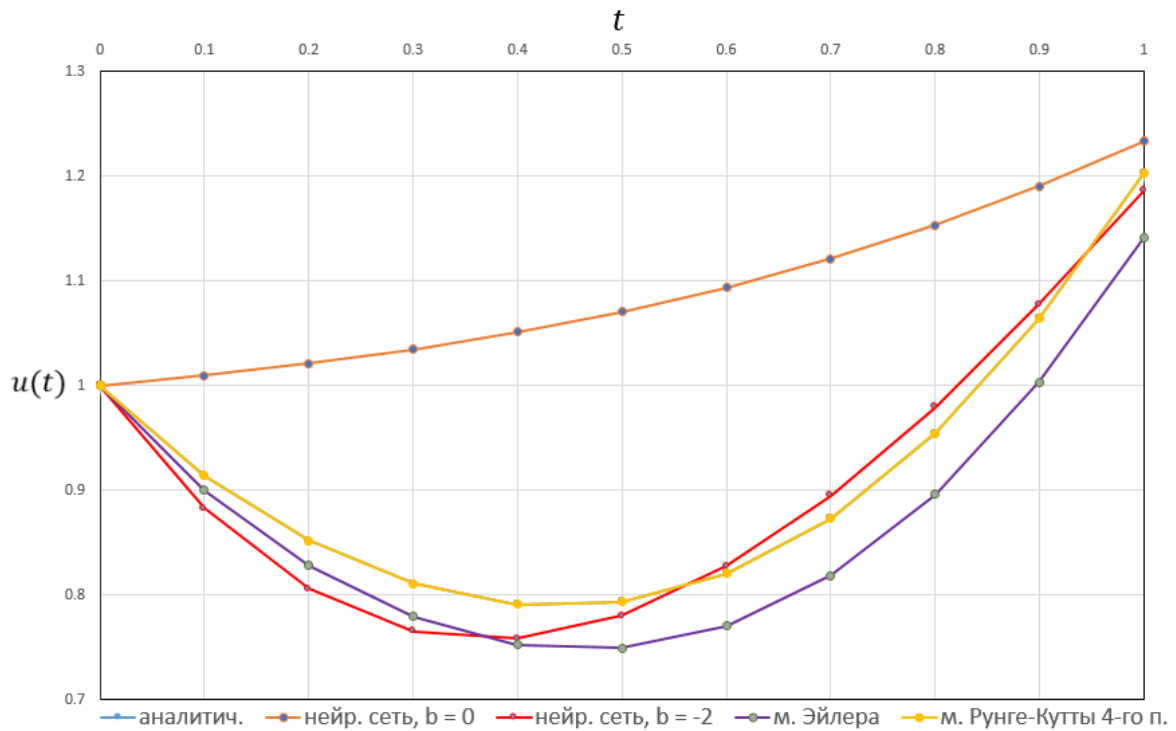


Рис. 14. Графики полученных решений.

4.4. Общий анализ

На рис 4, 5, 8, 11, 12 представлены логарифмические графики зависимости максимальной относительной погрешности (по сетке) и значения функции потерь от количества нейронов в скрытом слое. Действительно, выбранная функция потерь может являться мерой точности решения: при уменьшении относительной погрешности решения значение функции потерь так же склонно уменьшаться.

Необходимо отметить, что выбор b_{init} позволяет улучшить решение. Для сетки с шагом 0.1: в первом тесте $b_{init} = 0$ соответствует относительная погрешность 0.58761, $b_{init} = -1.49$ соответствует 0.02022; в третьем тесте $b_{init} = 0$ соответствует 0.34971, $b_{init} = -2$ соответствует 0.05583.

Важным параметром нейронной сети является количество нейронов в скрытом слое, так как на каждый дополнительный нейрон приходится 3 параметра, которые необходимо оптимизировать. Поэтому, чтобы сократить вычислительные затраты, нужно использовать сеть с наименьшим числом нейронов, необходимым для обеспечения требуемой точности решения. Эксперименты показали, что 4-х нейронов вместе с выбором b_{init} достаточно для обеспечения относительной погрешности (на сетке с шагом 0.1) 0.02022 для первого теста; 0.34959 для второго теста; 0.05583 для третьего теста.

Градиентный спуск с фиксированным шагом, равным 10^{-10} дал относительные погрешности результата: 0.01123 для первого теста; 0.20659 для второго теста; 0.14362

для третьего теста. Из этого можно сделать вывод, что выбор алгоритма минимизации позволяет улучшить решение.

В таблицах представлена зависимость абсолютной, относительной погрешностей от шага сетки, а также время, затраченное на получение решения нейронной сетью, методами Эйлера и Рунге-Кутты 4-го порядка. Точность рассматриваемого метода не зависит от выбора сетки, на которой происходит поиск решения. Метод требует большего времени выполнения расчетов по сравнению с методами Эйлера и Рунге-Кутты 4-го порядка.

5. Заключение

В работе рассмотрен метод решения задачи Коши для ОДУ первого порядка с помощью искусственных нейронных сетей. Работа алгоритма показана на решении задач: $\frac{du}{dx} = -2u, u(0) = 2$; $\frac{du}{dx} = u^2x, u(0) = 1$; $\frac{du}{dx} = x^3 + 2x + x^2 \frac{(1+3x^2)}{1+x+x^3} - u \left(x + \frac{(1+3x^2)}{1+x+x^3} \right), u(0) = 1$. Рассмотрена зависимость относительной погрешности решения от количества нейронов в скрытом слое: экспериментальным способом установлено, что функция потерь вида $L[\vec{p}] = \sum_{i=1}^K \left\{ \frac{d\psi_t(x_i)}{dx} - f(x_i, \psi_t(x_i)) \right\}^2$ может являться мерой точности решения. Показано, что выбор начального значения результирующего нейрона и выбор алгоритма минимизации позволяет улучшить решение. Приведено сравнение с методами Эйлера и Рунге-Кутты 4-го порядка. Так же, установлено, что рассматриваемый метод требует большего времени выполнения, а также, что погрешность нейронной сети не зависит от выбора сетки, на которой ищется решение. Минимальная полученная относительная погрешность: 0.01123 для первого теста; 0.20659 для второго теста; 0.14362 для третьего теста.

К достоинствам рассмотренного метода относится то, что погрешность нейронной сети не зависит от выбора сетки, на которой ищется решение. К недостаткам относится сложность реализации и большее время, необходимое для получения решения (в сравнении с методом Рунге-Кутты 4-го порядка и методом Эйлера).

Список литературы

1. Artificial neural networks for solving ordinary and partial differential equations, I. E. Lagaris, A. Likas and D. I. Fotiadis, Department of Computer Science, University of Ioannina, Ioannina, Greece, 1997.
2. Solving differential equations using neural networks, M. M. Chiaramonte, M. Kiener, 2013.