

УДК 004.852
DOI: 10.15827/0236-235X.124.673-676

Дата подачи статьи: 24.07.18
2018. Т. 31. № 4. С. 673–676

Исследование комбинированного алгоритма при обучении трехслойных нейронных сетей различной топологии

*Е.Н. Остроух*¹, к.т.н., доцент, eostr@donpac.ru

*А.Н. Евич*¹, к.ф.-м.н., доцент, bkln@mail.ru; bkln@mail.ru

*Ю.О. Чернышев*¹, д.т.н., профессор, eostr@donpac.ru

*С.Д. Маркин*¹, магистрант, potate82@ya.ru

*П.А. Панасенко*², к.т.н., преподаватель, we_panasenko_777@mail.ru

¹ Донской государственной технической университет, г. Ростов-на-Дону, 344000, Россия

² Краснодарское высшее военное училище им. генерала армии С.М. Штеменко, г. Краснодар, 350063, Россия

При обучении нейронной сети настройка весовых коэффициентов осуществляется на основе минимизации ошибки вычислений. В случаях, когда целевая функция имеет сложный характер и обладает большим количеством локальных экстремумов, обучение сети с применением градиентных методов оптимизации зачастую не гарантирует нахождение глобального экстремума. Решение этой проблемы на сегодняшний день для большого класса задач осуществляется с помощью генетических алгоритмов как основного метода для обучения сетей обратного распространения ошибки. Развитие этих алгоритмов получило свое продолжение в изучении биоинспирированных алгоритмов и их гибридных модификаций. Использование биоинспирированных алгоритмов, основанных на методах случайного поиска, позволяет решить проблему обхода локальных экстремумов и обладает высокой скоростью сходимости.

В работе рассматривается комбинированный биоинспирированный алгоритм, реализующий решение задачи глобальной оптимизации в случае проблем, связанных с обучением искусственных нейронных сетей. Структура сети и число нейронов в каждом скрытом слое являются важными параметрами, влияющими на эффективность обучения искусственных нейронных сетей. Трехслойные нейронные сети позволяют решать большинство сложных задач. Однако влияние количества нейронов в каждом из скрытых слоев на скорость сходимости в общем случае мало изучено.

В работе исследуется ранее разработанная авторами комбинация алгоритма светлячков и градиентного спуска для обучения трехслойных нейронных сетей различной топологии.

Проведенное исследование дало возможность выявить из рассмотренных искусственных нейронных сетей топологию, позволяющую получить наиболее оптимальное решение за меньшее количество шагов.

Анализ производительности алгоритма обучения осуществлен на основе функции «исключающее ИЛИ» (Xor).

Ключевые слова: алгоритмы оптимизации, нейронная сеть, алгоритм светлячков, градиентный спуск.

Нейронные сети сегодня позволяют успешно решать различные задачи, связанные с обработкой и анализом данных в области робототехники, медицины, экономики, связи, автоматизации производства и пр. К таким задачам можно отнести распознавание образов, формирование моделей принятия решений, системы управления различными устройствами, системы массового обслуживания и многие другие.

При решении такого рода задач получены различные результаты, в которых рассматривается большое количество конфигураций нейронных сетей с различными принципами их обучения [1–3]. Алгоритмы обучения *искусственных нейронных сетей* (ИНС) подразделяют на два класса: детерминистские и стохастические. В основе детерминистских методов при обучении сети лежит строгая последовательность действий, направленная на коррекцию весов сети на основе значений входных величин, выходов, полученных в результате расчетов, и желаемых выходов. В стохастических методах при обучении сети изменение весов основано на псевдослучайных значениях. При этом сохраняются значения, которые ведут к улучшению результата.

Одним из классических методов обучения многослойной сети является алгоритм обратного распространения ошибок [4]. Основная идея этого алгоритма базируется на градиентных методах оптимизации и применима только к дифференцируемым функциям активации нейронов сети. Минимизация среднеквадратичной ошибки сети осуществляется градиентным методом наискорейшего спуска. Обучение нейронной сети классическими алгоритмами зачастую приводит к попаданию в локальные минимумы. Различные комбинированные алгоритмы, обеспечивающие широту (диверсификацию) изменения величин весов, исследовались неоднократно [5–7]. Применение биоинспирированных алгоритмов при решении задач поисковой оптимизации с нелинейной целевой функцией обеспечивает диверсификацию поиска решений. Одним из таких методов является алгоритм светлячков (firefly algorithm, FA) [8]. Ранее авторами был рассмотрен алгоритм [9], в котором весовые коэффициенты нейронной сети уточняются на основе алгоритма светлячков. Выявлено, что использование этого алгоритма позволяет получить более эффективное решение по сравнению с классическим методом настройки весовых коэффициентов.

В то же время при обучении нейронных сетей количество нейронов и скрытых слоев влияет на быстродействие ее обучения, возникновение ошибок (в случае, если сеть не удалось обучить) и пр. Определение в общем случае этих параметров ИНС на сегодняшний день является до конца не изученной проблемой. Как правило, количество нейронов и скрытых слоев подбирается для каждой отдельной задачи [10–12]. В связи с этим большой интерес представляет вопрос о влиянии структуры нейронной сети на скорость и точность ее обучения.

В настоящем исследовании анализируется работа комбинированного алгоритма светлячков для настройки весовых функций для многослойных нейронных сетей с тремя входными нейронами, двумя скрытыми слоями, содержащими различное количество нейронов, и выходным слоем, содержащим один нейрон.

Описание алгоритма роя светлячков

В роли поисковых агентов выступают светлячки. В основе алгоритма лежит их поведение, наблюдаемое в живой природе. Для коммуникации между особями светлячки используют яркость света. Каждая особь характеризуется яркостью и позицией. При этом светлячки с наименьшей яркостью перемещаются к особям с наибольшей яркостью. Если яркость светлячков одинакова, они перемещаются произвольным образом. Яркость светлячка прямо пропорциональна величине целевой функции и обратно пропорциональна квадрату расстояния между особями:

$$I(r) = \frac{1}{1 + \tau \cdot r^2},$$

где r – расстояние до светлячка; τ – коэффициент поглощения света среды. Изменение позиции светлячка с номером i к светлячку с номером j осуществляется по формуле

$$X_i = X_j + e^{-\tau r_{ij}^2} (X_j - X_i) + \delta \cdot S,$$

где S – вектор, учитывающий изменение координаты перемещения светлячка; расстояние между светлячками с номерами i и j :

$$r_{ij} = \sqrt{2 \sum_{k=1}^n (r_i^k - r_j^k)^2}.$$

Алгоритм состоит из следующих шагов.

С1. Задаем начальные значения свободным параметрам алгоритма светлячков. Обозначим через N количество светлячков, через M – количество итераций.

С2. Случайным образом задаем n начальных точек, соответствующих начальным положениям светлячков в начальный момент времени $t = 0$, $X(0) = (X_1(0), X_2(0), \dots, X_n(0))$, в пространстве R^{X_i} .

С3. Вычисляем в этих точках значение целевой функции $f(X_i(0))$. Задаем значение счетчика числа итераций $k = 0$.

С4. Проверяем условие $f(X_i(k)) > f(X_j(k))$. Если условие истинно, передвигаем светлячка с номером j к светлячку с номером i . Новое расположение светлячка с номером j вычисляем по формуле

$$X_j(k) = X_j(k) + e^{-\tau r_{ji}^2} (X_i(k) - X_j(k)) + \delta \cdot S,$$

где δ – свободный параметр рандомизации; S – случайное число из интервала $(-1; 1)$.

С5. Вычисляем значение $f(X_j(k))$. Если $i < N$, полагаем $i = i + 1$, переходим к шагу С4. Если условие ложно, проверяем условие $j < N$. Если условие истинно, полагаем $j = j + 1$, $i = 1$ и переходим к шагу С4. Если $i = j = N$, завершаем алгоритм.

Настройка весовых коэффициентов искусственной нейронной сети светлячковым алгоритмом

Предлагаемый алгоритм обучения ИНС представлен на рисунке 1 и состоит из следующих шагов.

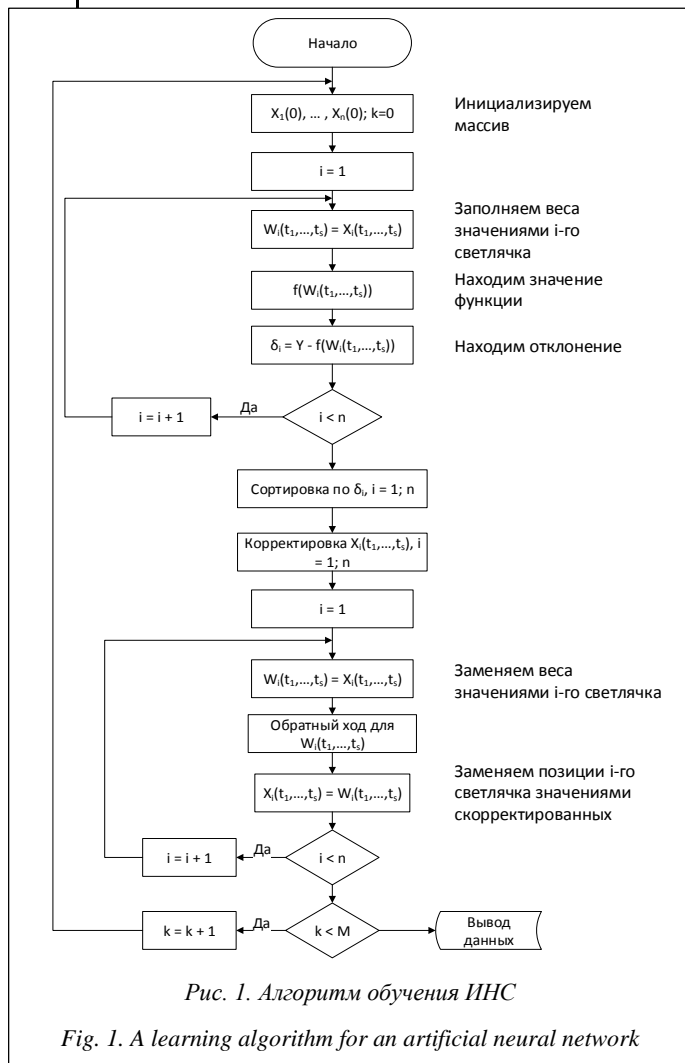


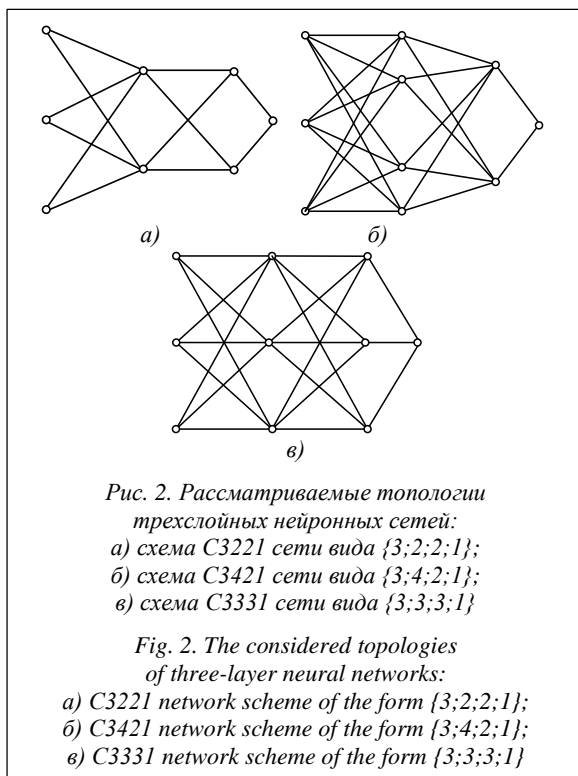
Рис. 1. Алгоритм обучения ИНС

Fig. 1. A learning algorithm for an artificial neural network

1. Выполняем шаги C1–C2 алгоритма светлячков.
2. Создаем ИНС с первоначальной инициализацией весовых коэффициентов w , соответствующих положениям светлячков в начальный момент времени.
3. Повторяем шаги 4–9.
4. На вход сети подаем входной вектор X_i , для которого получаем выходное значение Y_i .
5. Вычисляется вектор ошибки на выходе сети: $\delta_i = Y_i - Y$.
6. Выполняем шаг C4 алгоритма светлячков.
7. Вычисляем изменение вектора весовых коэффициентов в области малых ошибок по формуле $w(t + \Delta t) = w(t) + \gamma \cdot X_i \cdot \delta_i$, где γ – коэффициент скорости обучения ($0 < \gamma \leq 1$).
8. Проверяем условие завершения обучения. Обучение завершается, если суммарная абсолютная ошибка по всем векторам меньше ранее заданного малого значения.

Результаты экспериментальных исследований

Для исследования были взяты трехслойные сети (рис. 2), реализующие функцию Хог.



При тестировании алгоритма для всех видов сетей были использованы одни и те же параметры. Для алгоритма светлячков количество агентов в рое $N = 40$, количество циклов (эпох) алгоритма $t = 1$. Для обучения нейронных сетей скорость обучения 0,3, импульс 0,1, точность 0,000001.

Размерность D меняется в зависимости от структуры нейронной сети и соответствует количе-

ству связей в каждой сети. Результаты вычислительных экспериментов представлены на рисунках (<http://www.swsys.ru/uploaded/image/2018-4/2018-4-dop/10.jpg>, <http://www.swsys.ru/uploaded/image/2018-4/2018-4-dop/11.jpg>).

Выводы

В результате проведенных исследований из рассмотренных трехслойных моделей сетей была выявлена структура, позволяющая находить достаточно эффективное решение поставленной задачи. Такой структурой оказалась {3;4;2;1}. Дальнейшие исследования могут быть направлены на улучшение полученного решения нахождения глобального оптимума за счет дальнейшей настройки свободных параметров алгоритма обучения для выявленной структуры сети.

Статья написана при поддержке гранта РФФИ № 16-01-00391 «Разработка комбинированных алгоритмов для решения распределительных и транспортных задач с использованием идеологии искусственных иммунных систем и биоинспирированных алгоритмов».

Литература

1. Grossberg S. Nonlinear neural networks: Principles, mechanisms, and architectures. Neural networks, 1988, vol. 1, no. 1, pp. 17–61. DOI: 10.1016/0893-6080(88)90021-4.
2. Schmidhuber J. Deep learning in neural networks: An overview, Neural networks, 2015, no. 61, pp. 85–117. DOI: 10.1016/j.neunet.2014.09.003.
3. Каллан Р. Основные концепции нейронных сетей. М.: Вильямс, 2001. 288 с.
4. Rumelhart D.E., Hinton G.E., Williams R.J. Learning representations by back-propagating errors. Nature, 1986, vol. 323, no. 6088, pp. 533–536. DOI: 10.1038/323533a0.
5. Zhang J.R., Zhanga J., Lok Tat-Ming, Lyu M.R. A hybrid particle swarm optimization–back-propagation algorithm for feed-forward neural network training. Applied mathematics and computation, 2007, vol. 185, no. 2, pp. 1026–1037. DOI: 10.1016/j.amc.2006.07.025.
6. Ozturk C., Karaboga D. Hybrid artificial bee colony algorithm for neural network training. Evolutionary Computation (CEC), Proc. IEEE Congress, 2011, pp. 84–88. DOI: 10.1016/j.amc.2006.07.025.
7. Nawi N.M., Khan A., Rehman M.Z. A new back-propagation neural network optimized with cuckoo search algorithm. Proc. Intern. Conf. on Computational Science and Its Applications. Springer, Berlin, Heidelberg, 2013, pp. 413–426. DOI: 10.1007/978-3-642-39637-3_33.
8. Yang X.S. Firefly algorithms for multimodal optimization. Proc. Intern. Sympos. on Stochastic Algorithms. Springer, Berlin, Heidelberg, 2009, pp. 169–178. DOI: 10.1007/978-3-642-04944-6_14.
9. Остроух Е.Н., Евич Л.Н., Панасенко П.А. Разработка гибридного алгоритма решения оптимизационных задач принятия решений и управления // Искусственный интеллект: проблемы и пути их решения-2018: сб. докл. Междунар. конф. М.: Арсенал Отечества, 2018. № 1. С. 165–168.
10. Хайкин С. Нейронные сети: полный курс. М.: Вильямс, 2008. 1104 с.
11. Галушкин А.И. Нейронные сети. Основы теории. М.: Горячая линия–Телеком, 2012. 496 с.
12. Lin C.T., Lee C.S.G. Neural-network-based fuzzy logic control and decision system. IEEE Transactions on computers, 1991, vol. 40, no. 12, pp. 1320–1336. DOI: 10.1109/12.106218.

Investigation of a combined algorithm for learning three-layer neural networks of different topologies

E.N. Ostroukh¹, Ph.D. (Engineering), Associate Professor, eostr@donpac.ru

L.N. Evich¹, Ph.D. (Physics and Mathematics), Associate Professor, bkln@mail.ru; bkln@mail.ru

Yu.O. Chernyshev¹, Dr.Sc. (Engineering), Professor, eostr@donpac.ru

S.D. Markin¹, Graduate Student, potate82@ya.ru

P.A. Panasenko², Ph.D. (Engineering), Lecturer, we_panasenko_777@mail.ru

¹ Don State Technical University, Rostov-on-Don, 344000, Russian Federation

² Krasnodar Higher Military Engineering School, Krasnodar, 350063, Russian Federation

Abstract. When learning a neural network, the weighting factors are adjusted based on minimizing a calculation error. When the objective function has a complex character and a big number of local extremums, network learning using gradient optimization methods does not often guarantee the finding of a global extremum. Nowadays, the solution of this problem for a large class of problems includes using genetic algorithms as the main method for learning backpropagation networks. The development of these algorithms has continued in the study of bioinspired algorithms and their hybrid modifications. The use of bioinspired algorithms, which are based on random search methods, allows solving the problem of bypassing local extremums and has high convergence rate.

The paper considers a combined bioinspired algorithm that solves the global optimization problem when there are problems associated with learning artificial neural networks. The network structure and the number of neurons in each hidden layer are important parameters affecting the effectiveness of artificial neural networks learning. Three-layer neural networks can solve many complex problems. However, the effect of the number of neurons in each hidden layer on the convergence rate is under-explored in the general case.

The paper studies a combination of the firefly algorithm and gradient descent developed by the authors for the study of three-layer neural networks of various topologies. The conducted research made it possible to identify topology from artificial neural networks. This topology makes it possible to obtain the most optimal solution for fewer steps. The analysis of the learning algorithm performance is based on the exceptional-OR (Xor) function.

Keywords: optimization algorithms, neural network, firefly algorithm, gradient descent.

Acknowledgements. The research has been supported RFBR no. 16-01-00391 “Development of combined algorithms for solving distribution and transport problems using the ideology of artificial immune systems and bioinspired algorithms”.

References

1. Grossberg S. Nonlinear neural networks: Principles, mechanisms, and architectures. *Neural Networks*. 1988, vol. 1, no. 1, pp. 17–61. DOI: 10.1016/0893-6080(88)90021-4.
2. Schmidhuber J. Deep learning in neural networks: An overview. *Neural Networks*. 2015, no. 61, pp. 85–117. DOI: 10.1016/j.neunet.2014.09.003.
3. Kallan R. *Basic Principles of Neural Networks*. Moscow, Vilyams, 2001, 288 p.
4. Rumelhart D.E., Hinton G.E., Williams R.J. Learning representations by back-propagating errors. *Nature*. 1986, vol. 323, no. 6088, pp. 533–536. DOI: 10.1038/323533a0.
5. Zhang J.R., Zhanga J., Lok Tat-Ming, Lyu M.R. A hybrid particle swarm optimization–back-propagation algorithm for feedforward neural network training. *Applied Mathematics and Computation*. 2007, vol. 185, no. 2, pp. 1026–1037. DOI: 10.1016/j.amc.2006.07.025.
6. Ozturk C., Karaboga D. Hybrid artificial bee colony algorithm for neural network training. *IEEE Congress on Evolutionary Computation (CEC), 2011*. 2011, pp. 84–88. DOI: 10.1016/j.amc.2006.07.025.
7. Nawi N.M., Khan A., Rehman M.Z. A new back-propagation neural network optimized with cuckoo search algorithm. *Intern. Conf. on Computational Science and its Applications*. Springer, Berlin, Heidelberg Publ., 2013, pp. 413–426. DOI: 10.1007/978-3-642-39637-3_33.
8. Yang X. S. Firefly algorithms for multimodal optimization. *Intern. Symp. on Stochastic Algorithms*. Springer, Berlin, Heidelberg Publ., 2009, pp. 169–178. DOI: 10.1007/978-3-642-04944-6_14.
9. Ostroukh E.N., Evich L.N., Panasenko P.A. Development of a hybrid algorithm for solving optimization problems of decision making and control. *Artificial Intelligence: Problems and Solutions-2018: Proc. Intern. Conf.* Moscow, Arsenal Otechestva Publ., 2018, no. 1, pp. 165–168 (in Russ.).
10. Khaykin S. *Neural Networks: A Full Course*. Moscow, Vilyams Publ., 2006.
11. Galushkin A. I. *Neural Networks. Fundamentals of the Theory*. Moscow, Goryachaya liniya–Telekom Publ., 2012, 496 p.
12. Lin C.T., Lee C.S.G. Neural-network-based fuzzy logic control and decision system. *IEEE Trans. on Computers*. 1991, vol. 40, no. 12, pp. 1320–1336. DOI: 10.1109/12.106218.