

АЛГОРИТМЫ ОБУЧЕНИЯ НЕЙРОСЕТЕВОЙ СИСТЕМЫ СЧИСЛЕНИЯ ПУТИ СУДНА

Рассматривается методика определения оптимального по точности (сходимости) алгоритма обучения нейронной сети, прогнозирующей счислимые координаты судна. Данная методика основана на сравнении работы нескольких алгоритмов и реализована в два этапа. На первом этапе обучения определяется оптимальный алгоритм обучения при фиксированном числе итераций. Для каждого рассмотренного количества итераций алгоритм запускается по десять раз, и определяются характеристики точности прогноза сетью траектории. Для каждого числа итераций определяется среднее время обучения для наиболее точного алгоритма. Это время используется на втором этапе обучения, где оно задаётся одинаковым для всех алгоритмов. Они также запускаются по десять раз. Для каждого заданного значения времени определяется более точный алгоритм. Оптимальным по точности алгоритмом считается тот, который оказался самым точным для большинства моментов времени. На всех этапах основным критерием точности является наибольшее из десяти значений максимума модуля невязки на тестовой выборке.

Ключевые слова: нейронная сеть, счисление пути судна, алгоритм обучения, тестирование, координаты.

Введение

В настоящее время определение местоположения судна выполняется, главным образом, при помощи спутниковых навигационных систем, позволяющих получать координаты с высокой точностью и секундной дискретностью. Однако, несмотря на успехи в развитии технологий спутниковой навигации в последние десятилетия, её методы имеют существенный недостаток — неавтономность, что не позволяет использовать спутниковую систему при наступлении ряда известных обстоятельств. В связи с этим вопросы изучения автономных методов определения места судна — счисления — представляют практический интерес и в наши дни.

Погрешность счисления зависит от того, какой набор датчиков используется в качестве источника навигационной информации. Как правило, на судах торгового флота для определения счислимых координат применяются относительный лаг, измеряющий продольную составляющую скорости, и гирокомпас. В ряде случаев относительный лаг может измерять и поперечную составляющую относительной скорости. На крупнотоннажных судах устанавливается абсолютный лаг, измеряющий продольную и поперечную составляющие скорости судна. На некоторых кораблях определение счислимых координат возможно также и на основе инерциальной системы. Какой бы набор датчиков ни использовался, переход от измерений к координатам осуществляется, так или иначе, на основе определённого алгоритма. Этот алгоритм обязательно включает схему численного интегрирования параметров движения судна (ускорения или/и скорости), которая характеризуется определёнными методическими погрешностями. При использовании одноканального относительного лага в указанный алгоритм может быть включена также методика определения скорости (угла) дрейфа судна, характеризующаяся, в общем случае, погрешностями методического и теоретического плана.

Традиционные алгоритмы численного интегрирования обладают тем недостатком, что основаны они на замене подынтегральной функции её аппроксимацией, обычно линейной. В связи с этим представляет интерес разработка новых *нелинейных* методов обработки навигационной информации, не содержащих в своей основе предположений о конкретном виде интегрируемых параметров движения судна как функций времени. Построение подобных алгоритмов возможно на основе нейронных сетей, обладающих универсальными свойствами нелинейности при решении ряда прикладных задач [1], [2].

Варианты использования нейронных сетей для прогноза параметров движения судна описаны, например, в работах [3] – [5]. В данных исследованиях нейронная сеть выполняет функцию идентификации модели движения судна. Несколько иной подход к применению нейросетевых технологий для оценки навигационных параметров судна предлагается в статьях [6] – [8], где нейронные сети используются для прогноза траектории, а коррекция их свободных параметров выполняется в режиме реального времени по мере движения судна. В большинстве из указанных исследований не учитываются внешние факторы — ветер и волнение — при синтезе нейросетевых систем. Более того, проверка работоспособности нейронных сетей выполняется только для узких классов манёвров типа «циркуляция» и «зигзаг», а точность их в других навигационных ситуациях не исследуется. Использование же нейронной сети для определения счислимых координат судна предполагает её тестирование на множестве различных входных сигналов, которые могут иметь место в ходе эксплуатации судна.

Построение моделей счисления пути судна на основе нейронных сетей описывается, например, в работе [9], где предложены три варианта нейронных сетей, прогнозирующих скорость дрейфа судна в условиях внешних факторов. В исследовании [10] предлагается построение нейронной сети, прогнозирующей координаты судна на основе измерений относительного лага и гирокомаса. В [11], [12] предлагаются различные варианты синтеза нейросетевых систем счисления, прогнозирующих координаты судна (приращения к ним) при различных комплектациях датчиков автономной навигационной информации. В данных исследованиях использовались различные методы обучения нейросетевых систем. При этом выбор алгоритма обучения происходил, как правило, на основе общеизвестных рекомендаций и практики для решения подобного рода задач. В связи с этим интерес представляет исследование сходимости различных алгоритмов обучения конкретной нейросетевой системы счисления пути судна. Такой подход дает возможность определить алгоритм обучения, применение которого позволяет настроить нейронную сеть оптимальным, с точки зрения её точности, образом.

Архитектура нейронной системы счисления

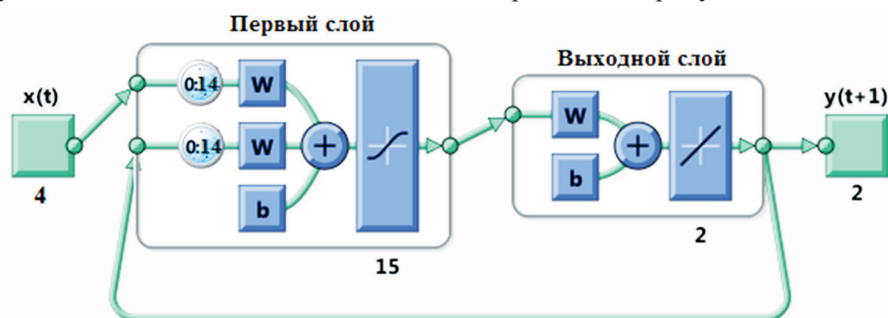
Исследование сходимости алгоритмов обучения будем выполнять для нейронной сети, прогнозирующей координаты судна по известным значениям его курса и составляющих относительной скорости [10]. Синтез и тестирование указанной сети рассматриваются также в [11], [12]. Входной вектор X имеет вид

$$X = (\sin K, \cos K, V_{ox1}, V_{oy1})^T,$$

где K — истинный курс судна; V_{ox1}, V_{oy1} — продольная и поперечная составляющие его относительной скорости.

Выходным сигналом Y служат координаты центра тяжести судна G в географической системе координат $Y = (x_G, y_G)^T$.

Сеть реализует модель нелинейной авторегрессии с внешними входами (NARX — *nonlinear autoregressive with exogenous inputs model*), имеет по 14 единичных задержек по линии как входного, так и выходного сигнала. Сеть двухслойная. Первый слой имеет 15 нейронов с сигмоидальными функциями активации (в виде гиперболического тангенса), второй — два нейрона с тождественными функциями активации. Общий вид сети приведен на рисунке.



Общий вид архитектуры нейронной сети

Архитектура и назначение нейронной сети во многом определяют область поиска возможных алгоритмов для её обучения. По причине того, что рекуррентная сеть NARX может рассматриваться как многослойный персептрон [2], основным методом её обучения является *метод обратного распространения ошибки*. В связи с этим возникает вопрос: «Какой режим обучения (последовательный или пакетный) будет для обучения рассматриваемой сети более предпочтительным?». Сеть обучается на основе *a priori* полученных данных, и её настройка в режиме реального времени не предполагается. Объёма оперативной памяти вполне достаточно для хранения информации об обучении. Более того, при обучении в пакетном режиме точная оценка вектора градиента достигается, как правило, без введения дополнительных процедур его оценки [13]. Наконец, некоторые алгоритмы (например, способ Левенберга–Марквардта) имеют лишь пакетную форму своей реализации. Указанные обстоятельства свидетельствуют более в пользу пакетного режима, чем режима последовательного.

Обзор основных алгоритмов обучения. Исторически первым алгоритмом обучения многослойных нейронных сетей прямого распространения методом обратного распространения можно считать *дельта-правило* (BP — backpropagation), которое для пакетного режима записывается следующим образом:

$$\mathbf{w}_{k+1} = \mathbf{w}_k - \eta \mathbf{g}_k, \quad (1)$$

где $\mathbf{w}$ — вектор свободных параметров сети (весов и пороговых значений) размерности P ; k — индекс итерации (эпохи); η — параметр скорости обучения; $\mathbf{g}$ — вектор частных производных функции стоимости $E(\mathbf{w})$ по свободным параметрам сети, определяющийся следующим выражением:

$$E(\mathbf{w}) = 0,5 \sum_{n=1}^N \sum_{m=1}^2 e_{n,m}^2, \quad (2)$$

где n — индекс образца из общего их числа N ; m — индекс выходного нейрона; $e_{n,m}$ — разность между прогнозируемым сетью значением сигнала n на m -м выходном нейроне и фактическим значением образца.

Вектор $\mathbf{g}$ имеет следующий вид: $\mathbf{g} = \left(\frac{\partial E}{\partial w_1}, \frac{\partial E}{\partial w_2}, \dots, \frac{\partial E}{\partial w_p} \right)$. Процедура расчёта частных про-

изводных детально описана в [2], [13], [14].

Одной из основных проблем, возникающих при реализации классического алгоритма обратного распространения, является решение вопроса о выборе значения параметра η . Общеизвестных методик его определения в настоящее время не существует. Тем не менее, разработан ряд методик, позволяющих корректировать величину η на каждой итерации обучения. В среде MATLABR2013b реализован следующий алгоритм, который можно назвать адаптивным дельта правилом (ABP — adaptive back propagation). Выражение (1) сохраняет полностью свой вид для указанного алгоритма за исключением того, что величина параметра скорости обучения имеет индекс итерации k :

$$\mathbf{w}_{k+1} = \mathbf{w}_k - \eta_k \mathbf{g}_k, \quad (3)$$

где η_k — адаптивный параметр скорости обучения, корректируемый в соответствии со следующей системой правил:

1) если от итерации к итерации значение функции стоимости $E(\mathbf{w})$ увеличивается в количество раз, большее $P_E = E_k / E_{k-1} > 1$, то коэффициенты сети не корректируются, а значение скорости обучения уменьшается умножением на величину $\lambda_d < 1$;

2) если от итерации к итерации значение функции стоимости $E(\mathbf{w})$ не уменьшается, но увеличение происходит в количество раз, меньшее $P_E = E_k / E_{k-1} > 1$, то коэффициенты сети обновляются;

3) если от итерации к итерации значение функции стоимости $E(\mathbf{w})$ уменьшается, то веса и пороги корректируются, а значение скорости обучения увеличивается умножением на величину $\lambda_{in} > 1$.

Конечно, описанный выше алгоритм адаптации параметра скорости обучения не является единственным. Широко известен, например, алгоритм *delta-delta* [14], в котором для каждого коэффициента сети на каждой итерации вычисляется свое значение параметра скорости обучения, учитывающее скорость изменения функции стоимости в направлении конкретного весового коэффициента. В работах [15], [16] предлагаются другие варианты настройки параметра скорости обучения.

Еще одним направлением модификации стандартного алгоритма ВР является добавление в его структуру инерционного члена α , описанное в работах [1], [2], [14], для увеличения скорости обучения без потери устойчивости. При этом существует проблема поиска оптимальных значений уже двух констант обучения (η и α) при решении конкретной задачи обучения.

Особенностью обучения рекуррентных нейронных сетей является проблема обращения градиента в нуль [13], связанная с тем, что «устаревшие» сигналы, стоящие в конце на линиях задержек, оказывают слабое влияние на значение прогнозируемого сетью выходного сигнала. Одним из способов уменьшения указанного эффекта в рекуррентных сетях является использование методов оптимизации второго порядка. В основу данных методов положена квадратичная аппроксимация поверхности ошибок в окрестности текущей точки [2], позволяющая записать уравнение настройки коэффициентов сети следующим образом [17]:

$$\mathbf{w}_{k+1} = \mathbf{w}_k - H_k^{-1} \mathbf{g}_k, \quad (4)$$

где H_k — матрица Гессе, содержащая вторые производные функции стоимости $E(\mathbf{w})$. Алгоритм коррекции (4) носит название *метода Ньютона*. Использование этого метода на практике затруднено в силу ряда причин. Во-первых, непосредственно процедура обращения матрицы H_k требует значительного объема вычислительных ресурсов. Во-вторых, по мере обучения нет гарантии, что обращение матрицы Гессе не будет представлять собой решение плохо обусловленной задачи, приводящее к усложнению вычислительного алгоритма. Наконец, если функция стоимости в окрестности текущей точки не является квадратичной, то гарантировать сходимость метода Ньютона невозможно.

Альтернативой методу Ньютона может служить способ *сопряженных градиентов*, при котором коррекция весовых коэффициентов осуществляется следующим образом:

$$\mathbf{w}_{k+1} = \mathbf{w}_k + \eta_k \mathbf{p}_k, \quad (5)$$

где величина $\mathbf{p}_k = -\mathbf{g}_k + \beta_k \mathbf{p}_{k-1}$ определяет направление поиска. Параметр η_k определяется на каждой итерации с использованием процедуры линейного поиска (метод золотого сечения, метод Брента и др.). Для вычисления параметра β_k используются различные формулы, среди которых одной из наиболее удачных с позиции сходимости является формула Полака–Рибьера [2]:

$$\beta_k = \max \left\{ \frac{\mathbf{g}_k^T (\mathbf{g}_k - \mathbf{g}_{k-1})}{\mathbf{g}_{k-1}^T \mathbf{g}_{k-1}}, 0 \right\}.$$

Значения величины $\mathbf{g}_k$ на каждой итерации k определяются на основе стандартного метода обратного распространения ошибки. Таким образом, рассматриваемый метод называется *методом сопряженных градиентов* на основе формулы Полака–Рибьера (*CGPR* — conjugate gradient Polak–Ribiere). Он содержит в себе, с одной стороны, свойства адаптивного метода обратного распространения и позволяет учитывать нелинейность функции стоимости в окрестности текущей точки — с другой. Причём эта нелинейность учитывается без непосредственного вычисления и обращения матрицы Гессе. Считается [2], что метод сопряжённых градиентов хорошо подходит для обучения многослойных сетей с большим количеством настраиваемых параметров, применяемых при решении задач аппроксимации функций, управления, регрессии.

Другим вариантом обхода затруднений, связанных с вычислением и обращением матрицы вторых производных, является использование *квазиньютоновских* методов. Уравнение обновления весов соответствует выражению (5), но направление поиска $\mathbf{p}_k$, в отличие от метода сопряжённых градиентов, уже определяется следующей формулой:

$$\mathbf{p}_k = -S_k \mathbf{g}_k, \quad (6)$$

где матрица S_k обновляется на каждой итерации по определённом алгоритму. Наиболее эффективным способом вычисления матрицы S_k считается в настоящее время [2] метод Бройдена–Флетчера–Гольдфарба–Шанно (*BFGS* — Broyden–Fletcher–Goldfarb–Shanno), при котором матрица S_k вычисляется следующим образом [13]:

$$S_k = S_{k-1} + \frac{\Delta \mathbf{w}_k \Delta \mathbf{w}_k^T}{\mathbf{q}_k^T \mathbf{q}_k} - \frac{S_{k-1} \mathbf{q}_k \mathbf{q}_k^T S_{k-1}}{\mathbf{q}_k^T S_{k-1} \mathbf{q}_k} + [\mathbf{q}_k^T S_{k-1} \mathbf{q}_k][\mathbf{v}_k \mathbf{v}_k^T],$$

$$\text{где } \mathbf{v}_k = \frac{\Delta \mathbf{w}_k}{\Delta \mathbf{w}_k^T \Delta \mathbf{w}_k} - \frac{S_{k-1} \mathbf{q}_k}{\mathbf{q}_k^T S_{k-1} \mathbf{q}_k},$$

а величины $\mathbf{q}_k$ и $\mathbf{w}_k$ определяются следующими выражениями:

$$\mathbf{q}_k = \mathbf{g}_k - \mathbf{g}_{k-1}; \quad \Delta \mathbf{w}_k = \mathbf{w}_k - \mathbf{w}_{k-1}.$$

Для определения параметра скорости обучения η_k , так же, как и в методе сопряженных градиентов, используется линейный поиск. Другой вариант аппроксимации матрицы Гессе используется в методе Левенберга–Марквардта (*LM* — Levenberg–Marquardt) [17]:

$$H \approx J^T J + \mu I,$$

где I — единичная матрица; μ — комбинационный коэффициент, а матрица Якоби J определяется следующим образом:

$$J = \begin{pmatrix} \frac{\partial e_{1,1}}{\partial w_1} & \frac{\partial e_{1,1}}{\partial w_2} & \dots & \frac{\partial e_{1,1}}{\partial w_p} \\ \frac{\partial e_{1,2}}{\partial w_1} & \frac{\partial e_{1,2}}{\partial w_2} & \dots & \frac{\partial e_{1,2}}{\partial w_p} \\ \dots & \dots & \dots & \dots \\ \frac{\partial e_{N,1}}{\partial w_1} & \frac{\partial e_{N,1}}{\partial w_2} & \dots & \frac{\partial e_{N,1}}{\partial w_p} \\ \frac{\partial e_{N,2}}{\partial w_1} & \frac{\partial e_{N,2}}{\partial w_2} & \dots & \frac{\partial e_{N,2}}{\partial w_p} \end{pmatrix}.$$

Для улучшения обобщающих свойств сети метод Левенберга–Марквардта нередко используется в комбинации со способом регуляризации Байеса (*LMBR* — Levenberg–Marquardt with Bayesian regularization). В работах [18], [19] приводятся примеры использования указанного метода. В алгоритме значение функции стоимости (целевой функции) F определяется следующим соотношением:

$$F = \beta E + \alpha E_w,$$

где E_w — сумма квадратов весовых коэффициентов сети; α, β — параметры регуляризации, обновляемые после каждой итерации алгоритма Левенберга–Марквардта.

Эксперимент. Перейдем теперь к постановке компьютерного эксперимента по изучению сходимости различных алгоритмов обучения нейронной сети. Сделаем ряд предварительных замечаний. Эксперимент выполнялся в среде MATLAB R2013b, установленной на операционной системе Windows8, на компьютере с процессором Intel®Core™i7-4770K.

При обучении нейронной сети будем использовать две выборки: обучающую и тестовую, каждая из которых содержит 139700 образцов. Формирование первой описано в работе [10], второй — в работе [12]. Так как использовались генераторы случайных чисел, указанные выборки различаются.

В сравнительном исследовании участвуют следующие алгоритмы:

- адаптивный метод градиентного спуска (ABP);
- метод сопряжённых градиентов (CGPR);
- квазиньютоновский метод (BFGS);
- метод Левенберга–Марквардта (LM);
- метод Левенберга–Марквардта с регуляризацией Байеса (LMBR).

Начальное значение параметра скорости обучения $\eta_{k=0}$ для метода ABP было получено путем перебора этого параметра для классического алгоритма BP, который для каждого рассматриваемого значения (0.1, 0.01, 0.001 ... $1 \cdot 10^{-10}$) запускался по 10 раз. По критерию максимума модуля невязки на тестовой выборке было получено оптимальное значение η , равное 0.0000001. В качестве алгоритмов линейного поиска величины η_k использовались следующие методы: метод Чараламбу-са (method of Charalambous) для CGPR и метод поиска с возвратом (Backtracking method) для BFGS.

Сначала алгоритмы исследовались на сходимость при фиксированном числе итераций. Было рассмотрено 10, 50, 100, 150, 200, 250 и 300 итераций, для каждого количества которых обучение производилось по десять раз. На каждом этапе, соответствующем определенному значению итераций, фиксировались наибольшее $s_{\max}$ и среднее s_{mean} значения максимума модуля невязки на обучающей и тестовой выборках, а также среднее время обучения. Результаты представлены в табл. 1.

Таблица 1

Обучение при фиксированном числе итераций

Число итераций	Алгоритм обучения									
	ABP		CGPR		BFGS		LM		LMBR	
	Наибольшее / (среднее) значения максимума модуля невязки, мили									
	О	Т	О	Т	О	Т	О	Т	О	Т
10	4.45 (3.36)	4.38 (3.18)	3.65 (2.41)	3.44 (2.26)	5.46 (3.69)	5.61 (3.48)	3.72 (2.48)	3.41 (2.16)	1.78 (1.66)	1.36 (0.42)
Время, с	89		195		279		163		165	
50	6.09 (3.04)	4.70 (2.63)	2.00 (1.75)	1.65 (1.28)	5.24 (3.13)	4.25 (2.97)	3.78 (2.20)	3.58 (1.71)	1.92 (1.62)	1.27 (0.75)
Время, с	455		953		1456		762		769	
100	4.58 (2.51)	4.71 (2.31)	1.92 (1.76)	1.56 (0.89)	4.10 (2.47)	3.74 (2.52)	2.26 (1.76)	1.75 (0.82)	1.80 (1.56)	1.75 (0.48)
Время, с	883		1956		3005		1506		1533	
150	2.59 (2.04)	2.58 (1.82)	1.90 (1.74)	1.39 (0.61)	2.74 (2.17)	2.87 (1.67)	3.20 (1.72)	3.22 (1.09)	1.75 (1.20)	1.39 (0.55)
Время, с	1348		2984		4523		2250		2290	
200	2.68 (2.1)	2.90 (1.49)	2.14 (1.77)	1.84 (0.70)	2.18 (1.88)	2.91 (1.91)	1.91 (1.61)	1.77 (0.93)	1.86 (1.10)	1.04 (0.52)
Время, с	1810		3808		6005		3002		3069	
250	2.14 (1.92)	2.10 (1.45)	2.16 (1.90)	1.06 (0.59)	2.97 (2.03)	2.19 (1.48)	1.92 (1.52)	3.34 (1.22)	2.07 (0.77)	0.86 (0.50)
Время, с	2253		4828		7512		3758		3808	
300	2.42 (1.95)	2.13 (1.49)	3.07 (1.87)	1.63 (0.65)	2.49 (1.98)	2.63 (1.48)	1.82 (1.32)	1.77 (0.76)	1.07 (0.72)	1.71 (0.84)
Время, с	2702		5689		9058		4549		4577	

Примечание. Буква «О» относится к образцам обучающей выборки, «Т» — тестовой. В графе «Время» указано среднее время обучения.

Как видно из данной таблицы, по прошествии определённого количества эпох разные алгоритмы обучения имеют различную точность на тестирующей выборке. Меньшая величина $c_{\max}$ на образцах из данной выборки соответствует лучшему алгоритму обучения при заданном числе итераций. Теперь возникает следующий вопрос: «Какой алгоритм можно считать наиболее подходящим для решения рассматриваемой задачи обучения?» Оптимальным будем считать тот метод обучения, который демонстрирует лучшие по предельной точности (величине $c_{\max}$) результаты на тестирующей выборке в *большинстве* рассмотренных случаев. Каждый случай соответствует обучению с заданным числом итераций. С этой точки зрения оптимальным алгоритмом обучения является способ Левенберга–Марквардта в сочетании со способом регуляризации Байеса (LMBR).

Подобный подход к определению оптимального алгоритма обучения можно было бы считать совершенным, если бы все рассматриваемые алгоритмы имели одинаковое время выполнения одной итерации. Например, для пятидесятой итерации (см. табл. 1), наиболее точным оказался метод Левенберга–Марквардта на основе регуляризации Байеса (LMBR), и для прохода этих пятидесяти итераций в среднем требуется 769 с. Классический градиентный метод (ABP) оказался наименее точным, но самым быстрым (среднее время пятидесяти итераций 455 с). Т. е. метод ABP имеет практически двойной запас по времени по сравнению с алгоритмом LMBR. Возникает предположение, что за это время ABP может достичь точности, не худшей, а может быть, даже лучшей, чем LMBR. В связи с этим было принято решение повторить эксперимент, зафиксировав уже не число итераций, а *время* обучения. В качестве времени обучения на каждом итерационном этапе было выбрано среднее время обучения самого точного алгоритма для соответствующего числа итераций. На каждом временном этапе обучение выполняется также по 10 раз. Результаты представлены в табл. 2.

Таблица 2

Обучение при фиксированном времени

Время обучения, с	Алгоритм обучения									
	ABP		CGPR		BFGS		LM		LMBR	
	Наибольшее / (среднее) значения максимума модуля невязки, мили									
	О	Т	О	Т	О	Т	О	Т	О	Т
165	4.44 (3.17)	4.28 (2.97)	5.44 (2.71)	4.07 (2.34)	6.44 (4.25)	6.06 (3.94)	3.92 (2.60)	3.66 (2.14)	1.78 (1.62)	1.36 (0.49)
Число ит.	20		9		6		11		10	
769	5.92 (2.91)	4.27 (2.53)	2.06 (1.87)	1.81 (1.04)	3.38 (2.95)	3.23 (2.64)	4.53 (2.22)	4.17 (1.41)	1.92 (1.55)	1.61 (0.82)
Число ит.	91		39		27		51		50	
1956	3.67 (2.14)	3.74 (1.81)	2.15 (1.69)	1.32 (0.69)	4.24 (2.67)	4.28 (2.39)	1.95 (1.72)	1.60 (0.79)	1.95 (1.30)	1.61 (0.52)
Число ит.	222		100		66		131		129	
2290	2.23 (1.99)	2.20 (1.35)	1.86 (1.62)	1.03 (0.62)	3.09 (2.40)	3.66 (2.39)	1.85 (1.51)	2.01 (0.87)	1.75 (1.15)	1.40 (0.59)
Число ит.	258		117		76		154		152	
3069	2.55 (1.96)	2.21 (1.23)	2.84 (1.84)	1.44 (0.69)	2.28 (2.11)	2.81 (2.00)	2.06 (1.62)	1.49 (0.85)	1.93 (1.15)	1.15 (0.53)
Число ит.	346		157		102		206		204	
3808	2.77 (1.97)	2.07 (1.38)	2.80 (1.67)	1.43 (0.73)	3.17 (2.18)	2.35 (1.78)	1.92 (1.31)	2.87 (0.82)	2.07 (0.86)	0.86 (0.55)
Число ит.	430		198		126		255		254	
5689	2.08 (1.90)	1.97 (1.35)	2.79 (1.91)	2.48 (1.00)	2.20 (1.94)	2.40 (1.57)	1.79 (1.15)	1.66 (0.88)	0.99 (0.45)	1.71 (0.80)
Число ит.	641		306		192		379		381	

Примечание. Буква «О» относится к образцам обучающей выборки, «Т» — тестовой. «Число ит.» — среднее число итераций.

Как видно из табл. 2, наибольшую точность при фиксированном времени обучения имеет на большинстве временных этапов алгоритм LMBR, как это было и в предыдущем эксперименте с фиксированным числом итераций. Таким образом, данный алгоритм имеет наилучшие показатели по сходимости (величине $c_{\max}$) в обоих экспериментах. На втором месте стоит метод сопряжённых градиентов (CGPR), на третьем — метод Левенберга–Марквардта (LM). Таковы основные результаты выбора оптимального (по скорости сходимости) алгоритма обучения нейронной сети, прогнозирующей счислимы координаты судна по известным значениям его кинематических параметров. В пользу алгоритма Левенберга–Марквардта как одного из самых эффективных методов обучения нейронных сетей говорят и результаты исследований [17], [20].

Конечно, предлагаемая методика выбора оптимального алгоритма имеет определённые недочёты. Так, если сравнить число итераций наиболее точного алгоритма (см. табл. 1), по которому задаётся время обучения (см. табл. 2), со средним числом итераций (выделено курсивом в табл. 2), то заметно, что это их среднее значение в ряде случаев не совпадает с заданным числом из табл. 1. Погрешность не превышает 2 % от ожидаемого значения, но всё-таки говорит о некоторой нестабильности быстродействия вычислительной системы. Другой пример нестабильности следует из того, что алгоритм LM, имеющий среднее время одной итерации чуть меньшее, чем LMBR, успел в среднем пройти на две итерации меньше, чем LMBR на последнем временном этапе (см. табл. 2). Нестабильность может быть вызвана процессами, в различном количестве выполняющимися в разное время в операционной системе, температурой помещения, в которой установлен компьютер, и другими подобными программно-аппаратными эффектами.

Алгоритмом, имеющим наименьшее время выполнения одной итерации, справедливо оказался метод АБР, наибольшее — квазиньютоновский BFGS, повышенная вычислительная сложность которого (по сравнению с методами сопряжённых градиентов) отмечается в работах [2], [13]. Среднее время одной итерации алгоритма BFGS превышает таковое для CGPR в среднем на 53 %.

Заключение

Рассмотрена методика определения оптимального по точности (сходимости) алгоритма обучения нейронной сети, прогнозирующей координаты места судна по известным кинематическим параметрам. Данная методика основана на анализе точности прогноза траектории судна, обученной при помощи различных алгоритмов при фиксированном времени обучения. Методика включает два этапа. На первом этапе определяется оптимальный алгоритм обучения при фиксированных значениях числа итераций. Данные значения задаются дискретно с определённым шагом. На втором этапе фиксируется время обучения, устанавливаемое по среднему времени работы оптимального алгоритма (определённого на первом этапе) для каждого рассматриваемого числа итераций. Показателем точности выбирается наибольшее значение максимума модуля невязки в модельных ситуациях на образцах из тестовой выборки. Оптимальным по точности алгоритмом, определённым по указанной методике, оказался метод Левенберга–Марквардта в комбинации со способом регуляризации Байеса. Отбор алгоритмов, участвующих в сравнении, был сделан с учётом практики использования нейронных сетей для решения задач, связанных с моделированием динамики подвижных объектов. Дальнейшие исследования по разработке и тестированию алгоритмов обучения могут лежать в области изучения влияния количества свободных параметров сети и образцов на скорость сходимости процесса обучения, а также построения алгоритмов, учитывающих конкретный вид поверхности ошибок применительно к решаемой задаче счисления пути. Отдельного внимания заслуживают также вопросы разработки специализированных функций стоимости (целевых функций), отвечающих решению конкретных навигационных задач судовождения.

СПИСОК ЛИТЕРАТУРЫ

1. Каллан Р. Основные концепции нейронных сетей: пер. с англ. / Р. Каллан. — М.: Издат. дом «Вильямс», 2003. — 288 с.

2. Хайкин С. Нейронные сети: полный курс; пер. с англ. Н. Н. Куссуль, А. Ю. Шелестова / С. Хайкин. — М.: Издат. дом «Вильямс», 2006. — 1104 с.
3. Ведякова А. О. Идентификация в условиях внешнего возмущения с использованием нейронных сетей / А. О. Ведякова // International Journal of Open Information Technologies. — 2014. — Т. 2. — № 3. — С. 18–22.
4. Ebada A. Intelligent techniques-based approach for ship maneuvering simulations and analysis (Artificial Neural Networks Application): Doktor-Ing. genehmigte Dissertation / A. Ebada. — Germany: Institute of Ship Technology und Transport Systems, 2007. — 156 p.
5. Moreira L. Dynamic model of maneuverability using recursive neural networks / L. Moreira, C. Guedes Soares // Ocean Engineering. — 2003. — Vol. 30. — Pp. 1669–1697. DOI:10.1016/S0029-8018(02)00147-6.
6. Сазонов А. Е. Прогнозирование траектории движения судна при помощи нейронной сети / А. Е. Сазонов, В. В. Дерябин // Вестник Государственного университета морского и речного флота имени адмирала С. О. Макарова. — 2013. — № 3 (22). — С. 6–13.
7. Nguyen-H. M. Improving GPS/INS Integration through Neural Networks / M. Nguyen-H, C. Zhou // Journal of Telecommunications. — 2010. — Vol.2. — Is. 2. — Pp. 1–6.
8. Xu T. A Novel Approach for Ship Trajectory Online Prediction Using BP Neural Network Algorithm / T. Xu, X. Liu, X. Yang // Advances in information Sciences and Service Sciences (AISS). — 2012. — Vol. 4. — No. 11. — Pp. 271–277. DOI: 10.4156/AISS.vol4.issue11.33.
9. Дерябин В. В. Нейросетевые системы прогноза скорости дрейфа судна / В. В. Дерябин // Вестник Государственного университета морского и речного флота имени адмирала С. О. Макарова. — 2015. — № 5 (33). — С. 7–14.
10. Дерябин В. В. Нейросетевой метод прогноза счислимых координат места судна / В. В. Дерябин // Транспортное дело России. — 2014. — № 6. — С. 92–96.
11. Дерябин В. В. Прогноз счислимых координат судна на основе нейронных сетей / В. В. Дерябин // Транспортное дело России. — 2015. — № 4. — С. 159–165.
12. Дерябин В. В. Системы счисления пути судна на основе нейронных сетей / В. В. Дерябин // Транспортное дело России. — 2015. — № 5. — С. 137–142.
13. Haykin S. Neural Networks and Learning Machines / S. Haykin. — Third Edition. — New York: Prentice Hall, 2009. — 937 p.
14. Аникеев М. В. Обзор современных типов нейронных сетей / М. В. Аникеев, Л. К. Бабенко, О. Б. Макаревич // Радиоэлектроника, информатика, управление. — 2001. — № 1. — С. 48–56.
15. Mammadov M. An Improved Version of Backpropagation Algorithm with Effective Dynamic Learning Rate and Momentum / M. Mammadov, E. Tas // Proceedings of the 9th WSEAS International Conference on Applied Mathematics, Istanbul, Turkey, May 27 – 29. Turkey, 2006. — Pp. 356–361.
16. Subavathi S. J. Adaptive modified backpropagation algorithm based on differential errors // S. J. Subavathi, T. Kathirvalavakumar // International Journal of Computer Science, Engineering and Applications (IJCSEA). — 2011. — Vol. 1. — No. 5. — Pp. 21–34. DOI: 10.5121/ijcsea.2011.1503.
17. Yu H. Advanced Learning Algorithms of Neural Networks: Ph. D. dissertation / H. Yu. — USA: Graduate Faculty of Auburn University, 2011. — 130 p.
18. Foresee F. D. Gauss-Newton Approximation to Bayesian Learning / F. D. Foresee, M. T. Hagan // Neural Networks, 1997, International Conference on. — Houston, TX: IEEE, 1997. — Vol. 3. — Pp. 1930–1935. DOI: 10.1109/ICNN.1997.614194.
19. MacKay D. J. C. Bayesian Interpolation / D. J. C. MacKay // Neural Computation. — 1992. — Vol. 4. — No. 3. — Pp. 415–447. DOI:10.1162/neco.1992.4.3.415.
20. Седова Н. А. Интеллектуальная система автоматического управления судном по курсу: автореф. дис. ... канд. техн. наук: 05.22.19 / Н. А. Седова. — Владивосток, 2009. — 22 с.

TRAINING ALGORITHMS OF NEURAL NETWORK BASED VESSEL'S DEAD RECKONING SYSTEM

Methodology of optimal precision (convergence) neural network training algorithm determination is considered. Neural network predicts vessel's dead reckoning coordinates. The method based on several algorithms' performance comparison includes two phases. At the first phase optimal training algorithm is determined with

fixed number of iterations. For each of considered iteration number the training is performed ten times, and neural network track prediction precision characteristics are determined. For each number of epochs mean training time for most precise algorithm is registered. This time is used at the second phase, being defined the same for all algorithms considered. They are also performed ten times. For each set time of training value the most precise algorithm is determined. Algorithm, which demonstrates the best accuracy for the majority of time moments, is to be considered as optimal. At all stages the main accuracy criteria is maximum of ten values of track error maximum modules on test set.

Keywords: neural network, vessel's dead reckoning, training algorithm, testing, coordinates

REFERENCES

1. Kallan, Robert. *Osnovniye konseptsii neyronnyih setey*. M.: Williams publishing, 2003.
2. Haykin, Symon. *Neuronnye seti: polniy kurs*. M.: Williams publishing, 2006.
3. Vediakova, A. O. "Neural network approach for identification under external disturbance." *International Journal of Open Information Technologies* 2.3 (2014): 18–22.
4. Ebada, Adel. Intelligent techniques-based approach for ship maneuvering simulations and analysis (Artificial Neural Networks Application): Doktor-Ing. genehmigte Dissertation. Germany: Institute of Ship Technology und Transport Systems, 2007.
5. Moreira L., and C. Guedes Soares. "Dynamic model of maneuverability using recursive neural networks." *Ocean Engineering* 30.13 (2003): 1669–1697. DOI:10.1016/S0029-8018(02)00147-6.
6. Sazonov, A. E., and V. V. Deryabin. "Forecasting to paths of the motion ship with the help of neyronnoy network." *Vestnik Gosudarstvennogo universiteta morskogo i rechnogo flota imeni admirala S.O. Makarova* 3(22) (2013): 6–13.
7. Nguyen-H, Mathieu, and Chi Zhou. "Improving GPS/INS Integration through Neural Networks." *Journal of Telecommunications* 2.2 (2010): 1–6.
8. Xu, T., X. Liu, and X. Yang. "A Novel Approach for Ship Trajectory Online Prediction Using BP Neural Network Algorithm." *Advances in information Sciences and Service Sciences (AISS)* 4.11 (2012): 271–277. DOI: 10.4156/AISS.vol4.issue11.33.
9. Deryabin, V. V. "Neural network based vessel's speed of drift prediction systems." *Vestnik Gosudarstvennogo universiteta morskogo i rechnogo flota imeni admirala S.O. Makarova* 5(33) (2015): 7–14.
10. Deryabin, V. V. "Neural network method of vessel's dead reckoning position prediction." *Transport business in Russia* 6 (2014): 92–96.
11. Deryabin, V. V. "Vessel's dead reckoning track prediction on basis of neural networks." *Transport business in Russia* 4 (2015): 159–165.
12. Deryabin, V. V. "Vessel's dead reckoning systems on basis of neural networks." *Transport business in Russia* 5 (2015): 137–142.
13. Haykin, S. *Neural Networks and Learning Machines*. 3rd ed. New York: Prentice Hall, 2009.
14. Anikeev, M. B., L. K. Babenko, and O. B. Makarevich. "Obzor sovremennyih tipov neyronnyih setey." *Radioelektronika, informatika, upravlenie* 1 (2001): 48–56.
15. Mammadov, M., and E. Tas. "An Improved Version of Backpropagation Algorithm with Effective Dynamic Learning Rate and Momentum." *Proceedings of the 9th WSEAS International Conference on Applied Mathematics, Istanbul, Turkey, May 27-29. Turkey, 2006*: 356–361.
16. Subavathi, S. J., and T. Kathirvalavakumar. "Adaptive modified backpropagation algorithm based on differential errors." *International Journal of Computer Science, Engineering and Applications (IJCEA)* 1.5 (2011): 21–34. DOI: 10.5121/ijcsea.2011.1503.
17. Yu, Hao. *Advanced Learning Algorithms of Neural Networks*: Ph.D. dissertation. USA: Graduate Faculty of Auburn University, 2011.
18. Foresee, F. Dan, and Martin T. Hagan. "Gauss-Newton approximation to Bayesian learning." *Neural Networks, 1997. International Conference on*. Vol. 3. IEEE, 1997: 1930–1935. DOI: 10.1109/ICNN.1997.614194.
19. MacKay, David J. C. "Bayesian Interpolation." *Neural Computation* 4.3 (1992): 415–447. DOI:10.1162/neco.1992.4.3.415.

20. Sedova, N. A. *Intellektualnaya sistema avtomaticheskogo upravleniya sudnom po kursu: Abstract of PhD dissertation (Tech.)*. Vladivostok: Admiral Nevelskoy Maritime State University, 2009.

ИНФОРМАЦИЯ ОБ АВТОРЕ

Дерябин Виктор Владимирович —
кандидат технических наук, доцент.
ФГБОУ ВО «ГУМРФ имени
адмирала С.О. Макарова»
gmavitder@mail.ru, kaf_nav@gumrf.ru

INFORMATION ABOUT THE AUTHOR

Deryabin Victor Vladimirovich —
PhD, associate professor.
Admiral Makarov State University of Maritime
and Inland Shipping
gmavitder@mail.ru, kaf_nav@gumrf.ru

Статья поступила в редакцию 11 апреля 2016 г.

УДК 656.61

**Т. Е. Маликова,
А. А. Янченко**

ПРИМЕНЕНИЕ ТЕХНОЛОГИИ ПРЕДВАРИТЕЛЬНОГО ИНФОРМИРОВАНИЯ ТАМОЖЕННЫХ ОРГАНОВ ПРИ МОРСКИХ ВНЕПЛАНОВЫХ ГРУЗОПЕРЕВОЗКАХ

В статье рассмотрен процесс формирования информационного потока при внеплановых перевозках с целью определения необходимых условий для реализации технологии предварительного информирования таможенных органов на морском транспорте. В ходе исследования был выполнен структурный анализ процесса формирования импортного грузопотока при внеплановых перевозках. В результате выполненного исследования доказано, что механизм предоставления предварительной информации в отношении товаров, перемещаемых морским транспортом, разработан без учета специфических особенностей формирования грузопотока при внеплановых перевозках. Выявленные особенности технологического процесса формирования грузопотоков при внеплановых перевозках позволили выделить несколько вариантов развития событий, при наступлении которых предварительная информация от перевозчика и грузоотправителя либо не может быть сформирована, либо становится недостоверной, в результате чего судно не оформляется по технологии предварительного информирования. Для устранения выявленных проблем предложены мероприятия по изменению технологии предварительного информирования таможенных органов на морском транспорте.

Ключевые слова: контейнер, судно, перевозчик, морские перевозки, предварительная информация, таможенные органы.

Введение

В целях повышения пропускной способности морского порта за счет ускорения совершения таможенных операций в отношении морских судов и грузов ведутся исследования, направленные на разработку единого технологического процесса (ЕТП) взаимодействия всех участников транспортного рынка. В статье [1] изложены результаты натурного эксперимента по внедрению ЕТП в деятельность ОАО «Владивостокский морской торговый порт». В ходе экспериментальных исследований был сделан вывод о том, что существенное влияние на скорость движения контейнерного потока оказывает организация электронного документооборота в морском пункте пропуска с применением предварительного информирования контролирующих органов, прежде всего таможенных [2] – [4]. Предварительное информирование предусматривает выбор объектов таможенного контроля до фактического прибытия судна в порт [1], что позволяет сократить сроки пребывания грузов в порту под таможенным контролем и, таким образом, повысить пропускную способность терминалов порта. В результате применения данной технологии произошло сокращение срока фактического пребывания контейнера в морском порту Владивосток на 4,5 сут. Следует отме-