

Рисунок 3 .Прогнозируемый и эталонный ряды

Рассмотрим построение аттрактора Энона. Для этого отобразим полученные при помощи рекуррентной сети 300 значений ряда X на псевдофазовую плоскость (рисунок 4).

Как следует из рисунка, форма прогнозируемого аттрактора соответствует оригинальному аттрактору. Таким образом, рекуррентная нейронная сеть позволяет моделировать поведение нелинейных динамических систем.

ЗАКЛЮЧЕНИЕ

В работе рассмотрена технология применения рекуррентных нейронных сетей для прогнозирования и построения странных аттракторов. Для моделирования аттракторов использовалось отображение Энона. В дальнейшем планируется моделирование других хаотических процессов при помощи нейронных сетей.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Hornik K., Stinchcombe M., White H. Multilayer feedforward networks are universal approximators// *Neural Networks*, 1989, vol.2,359-366
2. Cubenko G. Approximation by superpositions of a sigmoidal function// *Mathematics of Control, Signals and Systems*, 1989, vol.2, N.4,303-314
3. Kim J., Kasabov N. HyFIS: adaptive neuro-fuzzy inference systems and their application to nonlinear dynamical systems// *Neural Networks*, 1999, vol.12, N.9, 1301-1319
4. Takens F. Detecting Strange Attractors in Fluid Turbulence. In *dynamical Systems and Turbulence*, 1981, Springer, Berlin
5. Шустер Г. Детерминированный хаос: Введение. – М.: Мир, 1988, 240 с.
6. Jordan M. Attractor dynamics and parallelism in a connectionist sequential machine// *proceedings of the 8 Annual Conference of the Cognitive Science*. – Hillsdale: Erlbaum – 1986, p.531-546
7. В.А. Головки. Нейроинтеллект: теория и применение. Книга 1: Организация и обучение нейронных сетей с прямыми и обратными связями. Брест., БПИ, 1999, 264 с.

УДК 681.324.01

Саченко А., Кочан В., Турченко В., Головки В., Савицкий Ю., Дунец А.

ПРОГНОЗИРОВАНИЕ ОШИБОК СЕНСОРНЫХ УСТРОЙСТВ С ИСПОЛЬЗОВАНИЕМ НЕЙРОННЫХ СЕТЕЙ

ВВЕДЕНИЕ

В подавляющем большинстве современных систем обработки сигналов сенсоров нормируется погрешность измерения выходного сигнала сенсора, а не физической величины. Например, при измерении температуры с помощью сенсоров Honeywell Pt100 [1] и блока Hydra 2625A Fluke [2] соотношение погрешностей элементов измерительного канала больше пятидесяти. Большинство работ по обработке сигнала сенсора [3, 4, 5], рассматривают вопросы, не связанные с повышением его точности. Надежное повышение точности сенсора независимо от условий эксплуатации обеспечивается его периодической поверкой по образцовому или калибровочной с помощью специального калибратора на месте эксплуатации, но операции, реализующие эти методы, достаточно трудоемки [6]. Снижение трудоемкости достигается прогнозированием дрейфа сенсора в межповерочный (межкалибровочный) ин-

тервал [7]. В этом случае наиболее эффективным является применение методов искусственного интеллекта, в частности нейронных сетей [8, 9, 10]. Известно [11], что качество обучения нейронных сетей в сильной степени зависит от объема данных, используемых для обучения. Это обуславливает основное противоречие, возникающее при использовании нейронных сетей для коррекции дрейфа сенсоров. Высококачественное обучение нейронной сети позволяет резко снизить погрешность прогноза и тем самым настолько увеличить межповерочный интервал, что полученное в результате поверок количество данных окажется недостаточным для высококачественного обучения нейронной сети.

Для разрешения этого противоречия, то есть для искусственного увеличения количества точек обучения прогнозирующей нейронной сети, предложено использовать «исторические» данные (данные о дрейфе однотипных сенсоров в

Саченко А.А. Директор института компьютерных информационных технологий Тернопольской академии народного хозяйства.

Турченко В. Аспирант каф. ЭВМиС Брестского государственного технического университета.

Дунец А.П. Аспирант каф. ЭВМиС Брестского государственного технического университета.

БГТУ, Беларусь, г. Брест, ул. Московская 267.

аналогичных условиях эксплуатации) [10] и дополнительную аппроксимирующую нейронную сеть [12]. Наиболее целесообразно совместное использование этих методов. Очевидно, что наилучшее качество прогноза обеспечивается обучением прогнозирующей нейронной сети на «реальных» данных о дрейфе (полученных в результате проверок или калибровок). Поэтому, по мере накопления «реальных» данных в процессе эксплуатации сенсора, они должны вытеснить «исторические» данные. Ниже рассмотрены особенности процесса искусственного увеличения количества точек обучения прогнозирующей НС предложенными выше методами.

1. МЕТОД ЗАМЕНЫ ИСТОРИЧЕСКИХ ДАННЫХ РЕАЛЬНЫМИ ДАННЫМИ

Пусть исторические данные о дрейфе сенсоров описываются кривыми $d_1...d_n$ (рисунок 1), которые в моменты a, b, c калибровок равны $d_i^a, d_i^b, d_i^c, i = \overline{1, n}$. Первая калибровка нового сенсора в момент a позволяет получить первое реальное значение d_k^a . Цель использования исторических данных – на основе d_k^a спрогнозировать ряд точек d_k^b, d_k^c и т.д. прогнозирующих значения дрейфа в моменты будущих калибровок. Для этого целесообразно использовать отдельную нейронную сеть, которая должна прогнозировать точку d_k^b на основе точек d_k^a и $d_i^a, i = \overline{1, n}$, следующую точку d_k^c на основе точек d_k^b и $d_i^b, i = \overline{1, n}$ и т.д. Количество имеющихся исторических кривых дрейфа сенсоров определяет структуру входного слоя нейронной сети.

Для прогноза целесообразно применить простейшую модель однослойного персептрона с линейной функцией активации нейрона. Выходное значение персептрона

$$y = \sum_{i=1}^n w_{i1} x_i - T, \quad (1)$$

где w_{i1} – весовые коэффициенты входов линейного нейрона, x_i – входные данные, T – порог нейрона.

Для обучения персептрона использовано правило Видроу-Хоффа [13]. Среднеквадратическая ошибка обучения сети

$$E = \sum_{k=1}^L E(k) = \frac{1}{2} \sum_{k=1}^L (y^k - d^k)^2, \quad (2)$$

где L – множество векторов обучения, $E(k)$ – среднеквадратическая ошибка сети для k -го входного вектора, y^k, d^k – выходное и желаемое значение для k -го входного вектора соответственно [14].

Конечные выражения для изменения весовых коэффициентов и порогов нейронной сети

$$w_{i1}(t+1) = w_{i1}(t) - \alpha(t)(y^k - d^k)x_i^k, \\ T(t+1) = T(t) + \alpha(t)(y^k - d^k) \quad (3)$$

где $i = \overline{1, n}, x_i^k$ – i -тый компонент k -го образа,

$$\alpha(t) = \left(1 + \sum_{i=1}^n x_i(t)^2 \right)^{-1} \quad (4)$$

– адаптивный шаг обучения.

Для обучения использован следующий алгоритм [14]:

1. Задать скорость обучения α ($0 < \alpha < 1$) и минимальную среднеквадратической ошибку сети E_{min} , которую необходимо достичь в процессе обучения;
2. Весовые коэффициенты и порог нейронной сети инициализировать случайным образом;

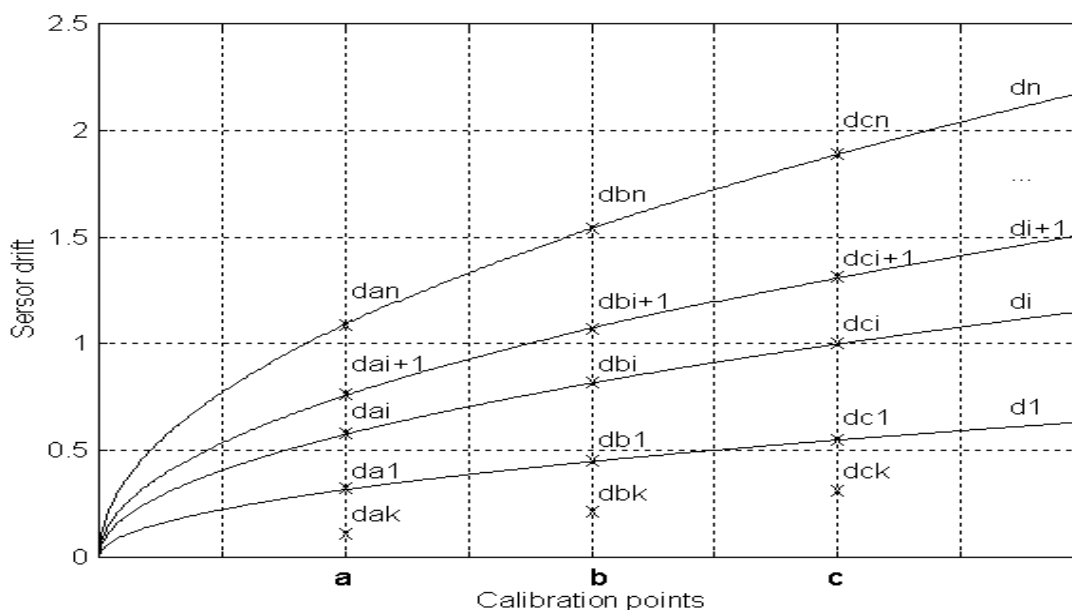


Рисунок 1. Исторические данные о дрейфе сходных типов сенсоров

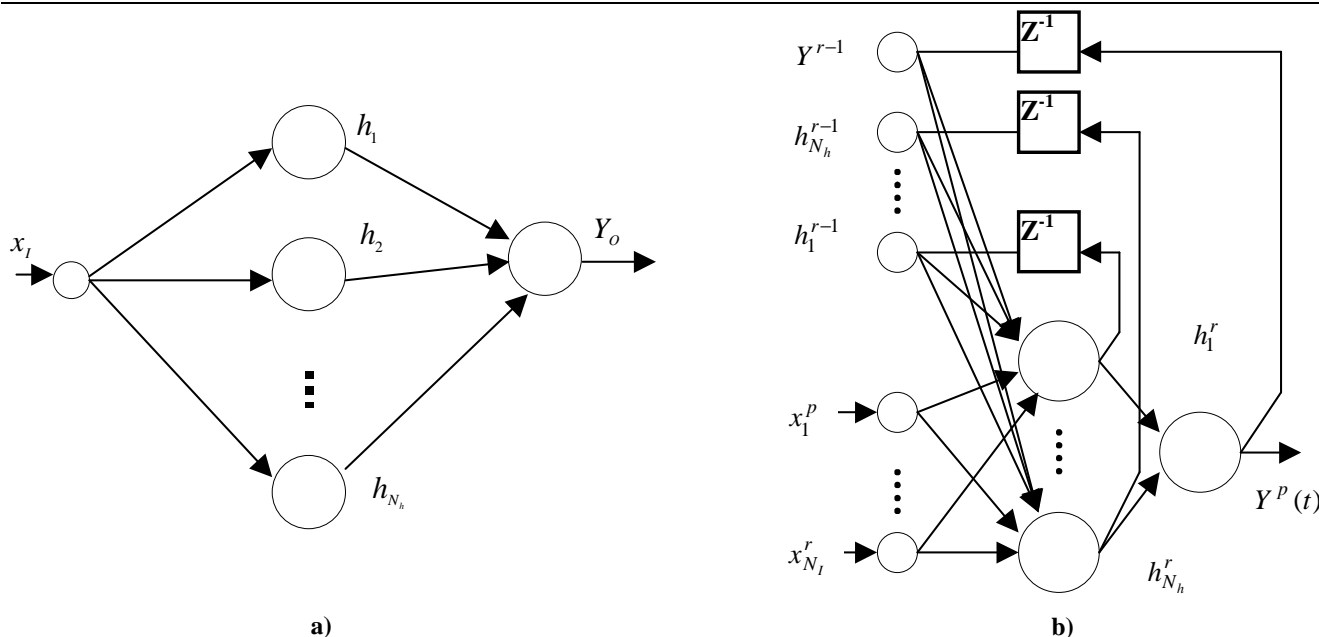


Рисунок 2. Структуры аппроксимирующей (а) и прогнозирующей (б) нейронных сетей

3. На вход нейронной сети подать входные данные, выходы вычислить согласно формуле (1);
4. Изменить значения весовых коэффициентов и порогов нейронной сети согласно с выражениями (3), (4);
5. Выполнять шаги 3-4 до тех пор, пока суммарная средне-квадратической ошибка сети (2) не станет меньше заданной, то есть $E \leq E_{min}$.

Прогнозирование дрейфа сенсора после калибровок b, c и т.д. осуществляется путем сдвига всего множества необходимых данных для формирования векторов обучения вправо на одну калибровку (см. рисунок 2).

2. МОДЕЛИ АППРОКСИМИРУЮЩЕЙ И ПРОГНОЗИРУЮЩЕЙ НЕЙРОННЫХ СЕТЕЙ

В качестве *аппроксимирующей* нейронной сети использован трехслойный персептрон, состоящий из входного распределительного слоя нейронов, одного скрытого слоя нейронов и одного линейного выходного нейрона (рисунок 2а). Выходная активность трехслойного персептрона

$$Y_o = \sum_{i=1}^{N_h} w_{i0} h_i - s_0 \quad (5)$$

где N_h - количество нейронов скрытого слоя, w_{i0} - веса от i -го нейрона скрытого слоя к выходному нейрону, h_i - их выходная активность, s_0 - порог выходного нейрона.

Выходное значение нейронов скрытого слоя

$$h_j = g(w_{ij} x_I + s_j), \quad (6)$$

где w_{ij} - веса от входного нейрона к нейрону j скрытого слоя, x_I - значение нейрона входного слоя, s_j - пороги скрытых нейронов. Для нейронов скрытого слоя используется сигмоидная функция активации

$$g(x) = \left(1 + e^{-x}\right)^{-1}. \quad (7)$$

Для обучения аппроксимирующей нейронной сети использован алгоритм обратного распространения ошибки [15]. Он

базируется на методе градиентного спуска в пространстве весовых коэффициентов и заключается в выполнении итерационной процедуры модификации весов и порогов нейронов для каждого эталона p обучающего множества

$$\Delta w_{ij}(t) = -\alpha \frac{E^p(t)}{w_{ij}(t)}, \Delta s_j(t) = -\alpha \frac{E^p(t)}{s_j(t)}, \quad (8)$$

где α - шаг обучения, $\frac{\partial E^p(t)}{\partial w_{ij}(t)}, \frac{\partial E^p(t)}{\partial s_j(t)}$ - градиенты

функции ошибки на итерации обучения t для обучающего эталона $p, p \in \{1, P\}, P$ - размер обучающего множества.

$$E^p(t) = \frac{1}{2} \left(Y_o^p(t) - D_o^p \right)^2,$$

где $Y_o^p(t)$ - выходная активность сети на итерации обуче-

ния t для обучающего эталона p, D_o^p - желаемое значение выходной активности сети для эталона p . Во время обучения происходит процесс снижения общей ошибки сети

$$E(t) = \sum_{p=1}^P E^p(t). \quad (9)$$

С целью улучшения параметров обучения нейронной сети и устранения недостатков классического алгоритма обратного распространения ошибки, связанных с эмпирическим выбором постоянного шага обучения, используем метод наискорейшего спуска для вычисления адаптивного шага обучения [12]. Таким образом, адаптивный шаг обучения для сигмоидной функций активации

$$\bar{\alpha}^p(t) = \frac{1}{(1 + (x_i^p)^2)} \times \frac{\sum_{j=1}^{N_h} (\gamma_j^p(t))^2 h_j^p(t) (1 - h_j^p(t))}{\left(\sum_{j=1}^{N_h} (\gamma_j^p(t))^2 (h_j^p(t))^2 (1 - h_j^p(t))^2 \right)} \quad (10)$$

где $\gamma_j^p(t)$ - ошибка нейрона j для сигмоидной функции для обучающего эталона p , и линейной функций активации

$$\alpha^p(t) = \frac{1}{\sum_{i=1}^{N_h} (h_i^p(t))^2 + 1} \quad (11)$$

где $h_i^p(t)$ - входные сигналы линейного нейрона в момент обучения t для обучающего вектора p .

$$\gamma_i^p(t) = \sum_{j=1}^{N_h} \gamma_j^p(t) w_{ij}(t) h_j^p(t) (1 - h_j^p(t)) \quad (12)$$

где $w_{ij}(t)$ - веса от нейронов скрытого слоя к выходному нейрону, адаптируемые на обучающей итерации t .

При организации обучения весьма актуальна проблема инициализации весовых коэффициентов нейронной сети. Скорость, точность и стабильность обучения зависит от этого в значительной степени. Обычно инициализация состоит в присвоении весам и порогам элементов случайных равномерно распределенных значений из некоторого диапазона: $w_{ij} = R(c, d)$, $s_j = R(c, d)$. Верхняя и нижняя границы этого диапазона определяются эмпирически. Мы предлагаем использовать математический подход для вычисления границ инициализации весов, что позволяет существенно сократить время обучения [12].

Для стабилизации процесса обучения использован алгоритм послонного обучения:

1. Вычислить верхнюю и нижнюю границы диапазона весов для нейронов скрытого слоя и для выходного нейрона, инициализировать веса и пороги нейронов;
2. Для тренировочного вектора p вычислить выходное значение сети $Y_o^p(t)$ используя (5, 6);
3. Вычислить ошибку $\gamma_o^p(t) = Y_o^p(t) - D_o^p$ выходного нейрона;
4. Модифицировать веса и пороги выходного нейрона (8) используя адаптивный шаг обучения (11);
5. Вычислить ошибку $\gamma_j^p(t)$ нейронов скрытого слоя для сети с модифицированными весами выходного слоя (12) для сигмоидной активационной функции (7);
6. Модифицировать веса и пороги нейронов скрытого слоя (8), используя адаптивный шаг (10) для сигмоидной активационной функции;
7. Выполнять шаги 2-6 до тех пор, пока общая ошибка обучения сети (9) не станет меньше заданной.

Применение данного алгоритма позволяет стабилизировать процесс обучения персептрона с различными функциями

активации нейронов и значительно уменьшить время обучения.

В качестве прогнозирующей использована трехслойная рекуррентная нейронная сеть, содержащая один скрытый слой нелинейных нейронов и один линейный выходной нейрон (рисунок 2б). Выходное значение нейронной сети

$$Y^r = \sum_{i=1}^{N_h} w_{io} h_i^r - s_o \quad (13)$$

где N_h - количество нейронов скрытого слоя, h_i^r - выходные значения нейронов скрытого слоя в момент r , s_o - порог активации выходного нейрона, w_{io} - веса от нейронов i скрытого слоя к выходному нейрону.

Выходное значение нейронов скрытого слоя момент времени r для тренировочного эталона p

$$h_j^r = g \left(\sum_{i=1}^{N_I} w_{ij} x_i^r + \sum_{k=1}^{N_h} w_{kj} h_k^{r-1} + w_{oj} Y^{r-1} + s_j \right) \quad (14)$$

где N_I - размерность входного вектора, w_{ij} - веса от вход-

ных нейронов i к нейронам j скрытого слоя, x_i^r - i -тый элемент входного вектора x^r , w_{kj} - веса от нейронов k до нейронов j скрытого слоя, $h_k^{r-1}(t)$ - выходная активность нейрона k в предыдущий момент времени $r-1$, w_{oj} - веса к нейронам скрытого слоя от выходного нейрона, $Y^{r-1}(t)$ - выходное значение сети в предыдущий момент $r-1$, s_j - пороги нейронов скрытого слоя.

В прогнозирующей нейронной сети в качестве функций активации нейронов скрытого слоя использованы сигмоидная функция (7) и нестандартная логарифмическая функция

$$g(x) = \ln \left(\frac{x + \sqrt{x^2 + a}}{\sqrt{a}} \right), \quad (a > 0) \quad (15)$$

Функция (15) является неограниченной на всей области определений. Это позволяет лучше моделировать и прогнозировать сложные нестационарные процессы. Параметр a определяет наклон функции активации. Для стандартной сигмоидной функции (7) адаптивный шаг обучения

$$\bar{\alpha}^p(t) = \frac{1}{\left(1 + \sum_{i=1}^{N_I} (x_i^p)^2 + \sum_{k=1}^{N_h} (h_k^{p-1}(t))^2 + (Y^{p-1}(t))^2 \right)} \times \frac{\sum_{j=1}^{N_h} (\gamma_j^p(t))^2 h_j^p(t) (1 - h_j^p(t))}{\left(\sum_{j=1}^{N_h} (\gamma_j^p(t))^2 (h_j^p(t))^2 (1 - h_j^p(t))^2 \right)} \quad (16)$$

где $\gamma_j^p(t)$ - ошибка нейронов j с сигмоидной функцией активации для обучающего эталона p

$$\gamma_i^p(t) = \sum_{j=1}^{N_h} \gamma_j^p(t) w_{ij}(t) h_j^p(t) (1 - h_j^p(t)) \quad (17)$$

Для логарифмической функции (15) адаптивный шаг обучения

$$\bar{\alpha}^p(t) = \frac{I}{\left(I + \sum_{i=1}^{N_I} (x_i^p)^2 + \sum_{k=1}^{N_h} (h_k^{p-1}(t))^2 + (Y^{p-1}(t))^2 \right)} \times \sqrt{a \sum_{j=1}^{N_h} (\gamma_j^p(t))^2 \left((B_j^p(t))^2 + a \right)^{-1}} \times \frac{1}{\left(\sum_{j=1}^{N_h} (\gamma_j^p(t))^2 \left((B_j^p(t))^2 + a \right)^{-1} \right)} \quad (18)$$

где

$$B_j^p(t) = \sum_{i=1}^{N_I} w_{ij}(t)x_i^p + \sum_{k=1}^{N_h} w_{kj}(t)h_k^{p-1}(t) + w_{0j}(t)Y^{p-1}(t) + s_j$$

взвешенная сумма входов нейронов j скрытого слоя, $\gamma_j^p(t)$ - ошибка нейрона j для обучающего эталона p , вычисляемая на каждой итерации обучения t :

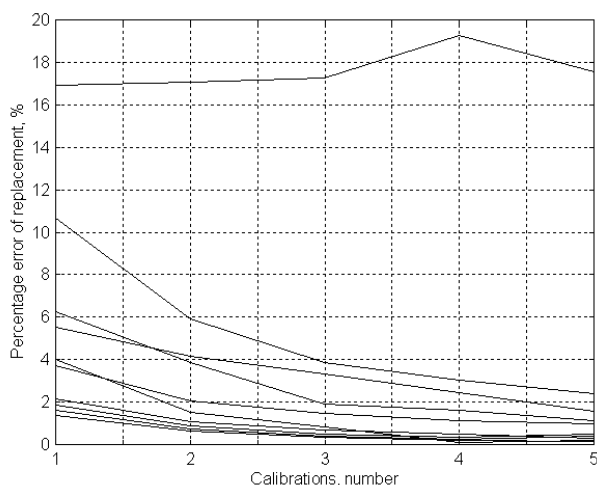
$$\gamma_i^p(t) = \sum_{j=1}^{N_h} \gamma_j^p(t)w_{io}(t) \frac{I}{\sqrt{I + (B_j^p(t))^2}} \quad (19)$$

$$\gamma_o^p(t) = Y_o^p(t) - D_o^p \quad (20)$$

В прогнозирующей нейронной сети также использован механизм адаптивной инициализации весов и порогов нейронной сети [12] и описанный выше алгоритм послыоного обучения, где вместо соответствующих выражений использовались формулы (13-20.)

4. ЭКСПЕРИМЕНТАЛЬНЫЕ ИССЛЕДОВАНИЯ

Векторы обучения для вытеснения данных (см. раздел 2) получены из 10 кривых семейства исторических дрейфов сенсоров. При обучении 10-ти входового однослойного перцептрона достигнута среднеквадратическая ошибка 10E-7. Все графики относительных погрешностей вытеснения исторических данных по 5 калибровкам представлены на рисунке 3а.



а)

Относительная погрешность вытеснения данных только в одном случае превышает 10%. Для оценки аппроксимирующей нейронной сети использованы 5 точек калибровок исторических данных. При обучении модели с 5 скрытыми нейронами и функцией активации (7) достигнута среднеквадратическая ошибка 2.4E-7. На рисунке 3б представлены максимальные и средние относительные погрешности аппроксимации в пяти точках калибровок. Максимальная погрешность не превышает 2%. Результат аппроксимации содержит 25 точек на каждой кривой исторических данных, на которых обучалась прогнозирующая нейронная сеть (модель с 10 входными нейронами, 10 скрытыми нейронами с функцией активации (7), и одним выходным линейным нейроном). Результаты прогноза (максимальные и средние относительные погрешности) для среднеквадратической ошибки обучения 7.8E-8 представлены на рисунке 3в. Как видно из рисунка, предлагаемые методы позволяют увеличить межкалибровочный интервал в 10 раз при допустимой погрешности прогноза 11%.

5. ЗАКЛЮЧЕНИЕ

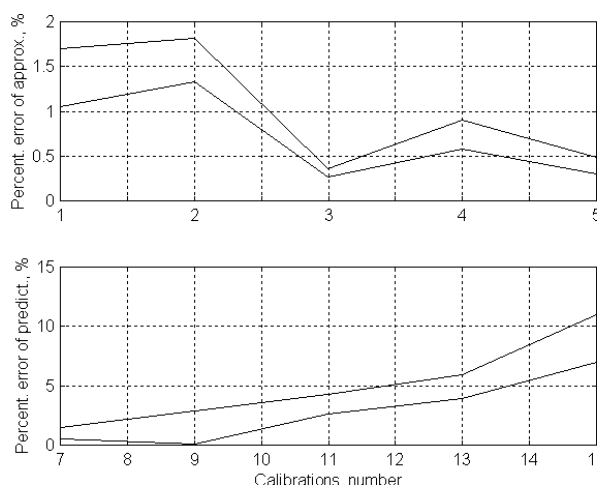
Особенности эксплуатации затрудняют прямое использование методов искусственного интеллекта, в частности, нейронных сетей для повышения точности систем обработки сигнала сенсора. Предложенные методы позволят обеспечить более резкое снижение погрешности измерения физической величины за счет высокой адаптивности в интеллектуальных системах. Эти методы лучше всего использовать в дистрибутивных иерархических системах [16, 17], в которых обучение нейронной сети ведется на верхнем уровне, а собственно прогнозирование – на нижних уровнях.

БЛАГОДАРНОСТИ

Авторы выражают признательность международной организации ИНТАС за финансовую поддержку, номер гранта INTAS-OPEN-97-0606.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. http://www.honeywell.com/sensing/prodinfo/temperature/catalog/c15_93.pdf
2. www.fluke.com/products/data_acquisition/hydra/home.asp?SID=7&AGID=0&PID=5308
3. Iyengar S.S., "Distributed Sensor Network - Introduction to the Special Section", *Transaction on Systems, Man and Cybernetics*, vol. 21, No. 5, 1991, pp 1027-1035
4. Finkelstein L., "Measurement and instrumentation centre",



б)

Рисунок 3. Процентные ошибки замены (а), аппроксимации и прогнозирования (б)

- Measurement, vol. 14, No 1, 1994, pp.23-29.
5. Sydenham P.H., "Sensing Science and Engineering Group", *Measurement*, vol 14, No 1, 1994, pp.81-87
 6. Brignell J., "Digital compensation of sensors", *Scientific Instruments*, vol. 20, No 9, 1987, pp.1097-1102
 7. Sachenko A., "Development of accuracy increasing methods and creation of precision systems of temperature measurement in industrial technologies", Dr. Techn. Sci. Thesis, Leningrad, 1988, 32p.
 8. C.Alippi, A.Ferrero, V.Piuri, "Artific.Intelligence for Instruments & Applications", *IEEE I&M Magazine*, Jun'98, pp.9-17.
 9. P. Daponte, D. Grimaldi, "Artificial Neural Networks in Measurements", *Measurement*, vol. 23, 1998, pp.93-115.
 10. Golovko V., Grandinetti L., Kochan V., Laopoulos T., Sachenko A., Turchenko V., "Sensor Signal Processing Using Neural Networks", *IEEE Region 8 Intern. Conf. Africon'99*, Cape Town (South Africa), 1999, pp.345-350.
 11. Kroese B., "An Introduction to Neural Networks", *Amsterdam*, University of Amsterdam, 1996, 120p.
 12. V.Golovko, J.Savitsky, A.Sachenko, V.Kochan, V.Turchenko, T.Laopoulos, L.Grandinetti, "Intelligent System for Prediction of Sensor Drift", *Proc. Intern. Conf. Neural Networks and Artificial Intelligence ICNNAI'99*, Brest (Belarus), 1999, pp.126-135.
 13. Widrow B., Hoff M., "Adaptive Switching Circuits", In 1960 *IRE WESCON Conv. Record, DUNNO*, 1960, pp.96-104.
 14. A.Sachenko, V.Kochan, V.Turchenko, V.Tymchyshyn, N.Vasylykiv, "Intelligent Nodes for Distributed Sensor Network", *Proceedings of 16th IEEE Instrumentation and Measurement Tech. Conf. IMTC/99*, Venice, Italy, 1999, pp.1479-1484.
 15. Golovko V.A., "NeuroIntelligence: Theory and Application. Vol. 1: Organisation and Learning of Neural Networks with Direct and Back Connections", Brest, *BPI*, 1999, 264p.
 16. Rumelhart D., Hinton G., Williams R., "Learning Representation by Backpropagation Errors", *Nature*, 1986, No 323, pp.533-536.
 17. A.Sachenko, V.Kochan, V.Turchenko, "Intelligent Distributed Sensor Network", *Proceedings of 15th IEEE Instrumentation and Measurement Tech. Conf. IMTC/98*, St. Paul (USA), vol.1., 1998, pp.60-66.
 18. V.Golovko, L.Grandinetti, V.Kochan, T.Laopoulos, A.Sachenko, V.Turchenko, V.Tymchyshyn, "Approach of an Intelligent Sensing Instrumentation Structure Development", *Proc. of IEEE International Workshop on Intelligent Signal Processing*, Budapest (Hungary), 1999, pp.336-341.

УДК 681.324.01

Головко В.А., Игнатюк О.Н.

ИСПОЛЬЗОВАНИЕ НЕЙРОСЕТЕВЫХ ТЕХНОЛОГИЙ ДЛЯ АВТОНОМНОГО УПРАВЛЕНИЯ МОБИЛЬНЫМ РОБОТОМ

ВВЕДЕНИЕ

Автономные мобильные роботы нуждаются в способности к обучению, для того, чтобы адаптироваться к изменяющейся внешней среде. Такое взаимодействие между роботом и внешней средой имеет большое значение для искусственных автономных систем. Жизнь полна ситуаций, которые невозможно предсказать. В этом случае способность робота к самообучению в процессе взаимодействия с динамически изменяющейся внешней средой позволяет ему адаптироваться к его окружению.

Самообучение характеризуется способностью системы обучаться при взаимодействии с внешней средой. В результате самообучения происходит самоорганизация системы с целью адаптации к внешней среде. Самообучение позволяет освободить оператора от процесса обучения и является важным фактором для эволюции системы. Так, при функционировании робота в разных условиях точность информации от сенсорных устройств может быть различной. В результате возникает необходимость корректировать знания, заложенные в систему, с целью адаптации к внешней среде. Особенно актуальным это является при функционировании робота в агрессивных средах или на других планетах, где невозможно предусмотреть все аспекты ситуационного взаимодействия робота с окружающей обстановкой.

В настоящее время существует два основных метода обучения искусственных автономных систем: обучение с учителем [1,2], при котором известна эталонная выходная информация, и подкрепляющее обучение [1,3,4], в котором имеется сигнал подкрепления. Во многих задачах управления получение тренировочных наборов затруднительно. Поэтому обучение с учителем редко используется в автономных системах в сравнении с подкрепляющим обучением.

В работе предлагается подход, который позволяет роботу формировать обучающую выборку в процессе взаимодей-

ствия с внешней средой. Такое обучение производится методом проб и ошибок. В качестве нейронных сетей используются в основном многослойные сети с прямыми связями, которые объединяются в интеллектуальную систему. Однако, здесь не имеется учителя. Робот начинает функционировать с небольшими априорными знаниями и в процессе взаимодействия с внешней средой может находить оптимальное поведение и генерировать знания. Это происходит в результате самообучения. Такой подход позволяет роботу успешно функционировать при отсутствии сведений о его динамической модели. Предлагаемый метод был апробирован путем моделирования. Результаты экспериментов приводятся.

1. ОБЩИЕ ПРИНЦИПЫ ОБОИДА ПРЕПЯТСТВИЙ

Рассмотрим основные принципы построения реактивной системы для управления мобильным роботом. В этом случае известны только координаты целевой точки. Задача робота состоит в том, чтобы достичь конечного пункта движения в неизвестном пространстве с препятствиями. Из этого следует, что знания в реактивной системе представляются в виде ситуация—реакция. При этом реакция соответствует текущему направлению движения робота. Задача робота состоит в нахождении кратчайшего пути между начальной и целевой точкой движения при отсутствии знаний относительно расположения препятствий.

1.1. ВХОДНАЯ ИНФОРМАЦИЯ

Входной информацией реактивной системы являются данные от различных сенсорных устройств. В результате обработки таких данных (data fusion) генерируется локальная карта местности, которая соответствует интеграционной картине окружающей обстановки. Такая карта характеризует в определенном радиусе обзора и угловом диапазоне 180 градусов расположение препятствий и расстояния до них: