

1490 of Lecture Notes in Computer Science. Springer Verlag, Berlin. 1998. P.391-410. **8.** Maleusieux F., Ridoux O., Boizumault P. Abstract compilation of Prolog. / In Jaar J. Editor // Joint International Conference and Symposium on Logic Programming. Manchester, United Kingdom. June 1998. MIT Press. P.130-144. **9.** Power J., Robinson E. Premonoidal categories and notions of computation. // Mathematical Structures in Computer Science. № 7(5). October 1997. P. 453-468. **10.** Corradini A., Asperti A. A categorical model for logic programs: Indexed monoidal categories. / In Proceedings REX Workshop '92 // Springer Lectures Notes in Computer Science. 1992. P. 5-36. **11.** Corradini A., Montanari U. An algebraic semantics for structured transition systems and its application to logic programs. // Theoretical Computer Science. №103(1). August 1992. P.51- 106. **12.** Barbuti R., Giacobazzi R., Levi G. A General Framework for Semantics-based Bottom-up Abstract Interpretation of Logic Programs // ACM Transactions on Programming Languages and Systems. № 15(1). 1993. P. 133-181. **13.** Finkelstein S., Freyd P., Lipton J. Logic programming in tau categories. // In Computer Science Logic '94, volume 933 of Lecture Notes in Computer Science. Springer Verlag, Berlin. 1995. P. 249-263. **14.** Шекета В.І. Модифікаційні предикатні запити / Науковий журнал «Проблеми програмування» інституту програмних систем НАН України. 2004. №2-3. С.339-343

// Спеціальний випуск за матеріалами 4-ї МНПК "УкрПроґ'2004", 1-3 червня 2004. Київ, Кібернетичний центр НАН України. **15.** Шекета В.І. Ініціалізація еластичних семантик над простором Гербранда для модифікаційних предикатних запитів // Міжнародний науково-технічний журнал «Вимірювальна та обчислювальна техніка в технологічних процесах». Хмельницький. 2003. №2(22). С.13-18. **16.** Шекета В.І. Аналіз семантики шаблонів виклику модифікаційних предикатних запитів для інформаційних систем на основі баз даних і знань // Комп'ютерна інженерія та інформаційні технології // Вісник національного університету "Львівська політехніка". Львів. 2003. № 496. С.217s228.

Надійшла до редколегії 12.09.2005

Рецензент: д-р техн. наук, проф. Горбійчук М.І.

Шекета Василь Іванович, канд. техн. наук, доцент кафедри програмного забезпечення факультету автоматизації та комп'ютерних наук Івано-Франківського національного технічного університету нафти і газу. Наукові інтереси: абстрактне логічне програмування, інформаційні системи на основі баз даних і знань. Адреса: Україна, 76019, м. Івано-Франківськ, вул. Карпатська, 15, тел.: (380) 03422 4-21-27 (роб.), e-mail: sheketa@mail.ru.

УДК 681.513:519.7

МНОГОМЕРНАЯ ИСКУССТВЕННАЯ НЕЙРОННАЯ СИГМА-ПИ СЕТЬ И АЛГОРИТМ ЕЕ ОБУЧЕНИЯ

БОДЯНСКИЙ Е.В., КУЛИШОВА Н.Е.

Предлагается многомерная искусственная нейронная сигма-пи сеть, позволяющая восстанавливать с заданной точностью произвольную многомерную функцию векторного аргумента. Отличительной особенностью сети является одновременное использование как радиально-базисных, так и сигмоидальных активационных функций. Предложенный градиентный алгоритм обучения основан на обратном распространении ошибки и позволяет настраивать синаптические веса сети в реальном времени. Результаты эксперимента не противоречат теоретическим.

Введение

Для решения задач моделирования, идентификации и управления широко применяются искусственные нейронные сети, в частности, многослойные персептроны (MLP) и радиально-базисные сети (RBFN) [1-4]. Многослойные персептроны весьма эффективны как универсальные аппроксиматоры [5]. По аппроксимирующим свойствам им не уступают и RBFN [6], однако низкая скорость обучения MLP, основанного на обратном распространении ошибок, ограничивает их применение, особенно в задачах реального времени. Основным же недостатком RBFN является экспоненциальный рост количества нейронов с увеличением размерности вектора входных сигналов, так называемое «проклятие размерности».

Обобщение положительных свойств MLP и RBFN обеспечивают Σ -П нейронные сети [2]. Они организованы из двух слоев нейронов, причем нейроны

скрытого слоя имеют нелинейные функции активации двух типов (сигмоидальные и радиально-базисные). Выходной слой осуществляет линейную комбинацию выходных сигналов нейронов скрытого слоя. Хотя подобная архитектура позволяет Σ -П сети аппроксимировать практически любые функции [7], на практике это свойство реализовано в отношении систем с отображением $\mathbf{R}^n \rightarrow \mathbf{R}^1$ [8-11].

Целью исследований является разработка архитектуры Σ -П сети, эффективной для интерполяции и аппроксимации многомерных функций.

1. Архитектура сети

Большинство реальных объектов характеризуется многомерными совокупностями входных и выходных параметров, в связи с чем для моделирования подобных объектов предлагается многомерная Σ -П сеть, архитектура которой представлена на рис. 1.

Сеть имеет n входов в нулевом слое, содержит h пар нейронов в скрытом слое, m нейронов в выходном слое и реализует отображение $y \in \mathbf{R}^n \rightarrow x \in \mathbf{R}^m$ в форме $\hat{y} = F(x)$.

Вектор входной последовательности $(n+1)$ поступает на скрытый слой, который состоит из двух блоков нейронов с разными функциями активации.

Каждому нейрону скрытого слоя предшествует сумматор с $n+1$ настраиваемыми входами каждый. Парно нейроны объединены умножителями. Выходной слой образован m нейронами типа адаптивного линейного ассоциатора с $h+1$ входами каждый. Всего же сеть содержит $h(2n+m+2)+m$ настраиваемых параметров, подлежащих восстановлению в процессе обучения.

Пусть нелинейный объект обладает неизвестной передаточной функцией такой, что

$$\mathbf{y}(k) = \mathbf{f}(\mathbf{x}(k)), \quad (1)$$

где $\mathbf{x}(k)$ – n -мерный вектор входных переменных; $\mathbf{y}(k)$ – m -мерный вектор выходных переменных.

Тогда выходной сигнал, формируемый сетью, соответствующей объекту (1) и содержащей h нейронов, можно записать в форме

$$\begin{aligned} \hat{\mathbf{y}}(k) &= \mathbf{F}(\mathbf{x}(k)) = \mathbf{w}_0(k) + \mathbf{w}(k)\tilde{\mathbf{y}}(k) = \\ &= \mathbf{w}_0(k) + \mathbf{w}(k)[\varphi(\mathbf{v}(k)) \circ \psi(\mathbf{u}(k))] = \\ &= \mathbf{w}_0(k) + \mathbf{w}(k)[\varphi_i(\mathbf{a}(k)\mathbf{x}(k)) \circ \psi_i(\mathbf{b}(k)\mathbf{x}(k))], \quad (2) \end{aligned}$$

здесь

$$\mathbf{w}(k) - h \times m, \mathbf{a}(k) - h \times (n+1), \mathbf{b}(k) - h \times (n+1)$$

– матрицы настраиваемых синаптических весов; $\mathbf{w}_0(k)$ – m -мерный вектор синаптических весов; $\varphi(\bullet)$ – радиально-базисные функции активации; $\psi(\bullet)$ – сигмоидальные функции активации; $\circ$ – символ прямого произведения векторов.

Введя

$$\mathbf{v}_i(k) = \sum_{j=0}^n a_{ij}(k)x_j(k) = \mathbf{a}_i^T(k)\mathbf{x}(k),$$

$$\mathbf{u}_i(k) = \sum_{j=0}^n b_{ij}(k)x_j(k) = \mathbf{b}_i^T(k)\mathbf{x}(k)$$

– входные векторы для нейронов скрытого слоя (здесь $\mathbf{a}_i^T$ – i -я строка $h \times (n+1)$ матрицы весов $\mathbf{a}(k)$; $\mathbf{b}_i^T$ – i -я строка $h \times (n+1)$ матрицы весов $\mathbf{b}(k)$), можно переписать (2) в форме:

$$\begin{aligned} \hat{\mathbf{y}}(k) &= \mathbf{w}_0(k) + \mathbf{w}(k)[\varphi_i(\mathbf{a}(k)\mathbf{x}(k)) \circ \psi_i(\mathbf{b}(k)\mathbf{x}(k))] = \\ &= \mathbf{w}_0(k) + \mathbf{w}(k)[\varphi_i(\mathbf{v}_i(k)) \circ \psi_i(\mathbf{u}_i(k))]. \quad (3) \end{aligned}$$

В качестве активационных функций, удовлетворяющих всем необходимым требованиям по точности аппроксимации [6], можно выбрать традиционный гауссиан и гиперболический тангенс

$$\varphi(u) = e^{-\frac{u^2}{\sigma}}, \quad \psi(u) = \tanh(u),$$

а также более простые тригонометрические функции

$$\varphi(u) = \cos(u), \quad \psi(u) = \sin(u).$$

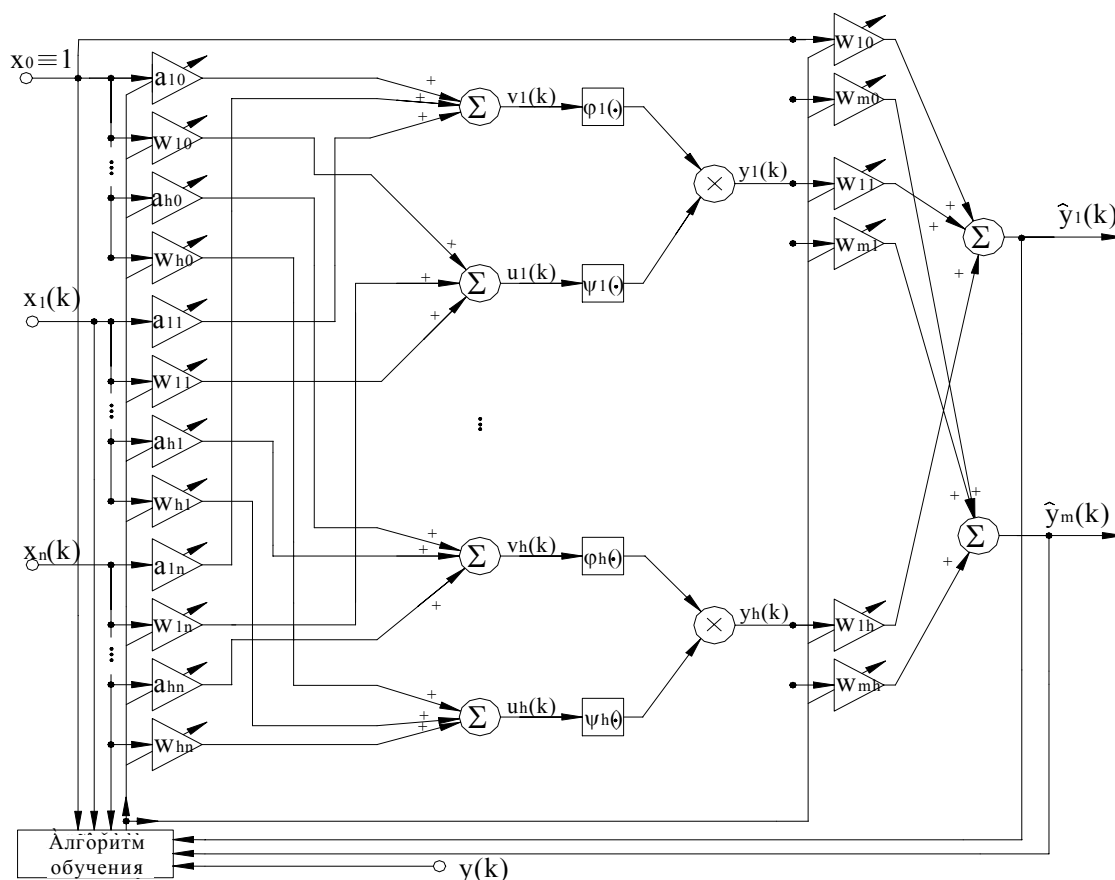


Рисунок 1. Архитектура многомерной сигма-пи нейронной сети

2. Алгоритм обучения

Для настройки матрицы весов $\mathbf{w}(k)$ используется одношаговый критерий обучения

$$\mathbf{E}(k) = \frac{1}{2} \|\mathbf{e}(k)\|^2 = \frac{1}{2} \|\mathbf{y}(k) - \tilde{\mathbf{y}}(k)\|^2. \quad (4)$$

Выходной слой образован адаптивными линейными ассоциаторами, а их обучение проводится путем минимизации критерия $\mathbf{E}(k)$ в соответствии с известным алгоритмом Уидроу-Хоффа [2]:

$$\mathbf{w}(k+1) = \mathbf{w}(k) + \frac{(\mathbf{y}(k+1) - \mathbf{w}(k)\tilde{\mathbf{y}}(k+1))\tilde{\mathbf{y}}^T(k+1)}{\|\tilde{\mathbf{y}}(k+1)\|^2}, \quad (5)$$

где $\tilde{\mathbf{y}}(k+1) = (1, \mathbf{y}^T(k+1))^T - (h+1) \times 1$ вектор выходных сигналов нейронов скрытого слоя.

Алгоритм обучения можно записать в виде системы:

$$\begin{cases} \mathbf{a}_i(k+1) = \mathbf{a}_i(k) - \eta_a(k) \nabla_{\mathbf{a}_i(k)} \mathbf{E}(k), \\ \mathbf{b}_i(k+1) = \mathbf{b}_i(k) - \eta_b(k) \nabla_{\mathbf{b}_i(k)} \mathbf{E}(k), \end{cases}$$

где $\eta_a(k), \eta_b(k)$ – параметры шага обучения матриц $\mathbf{a}(k)$ и $\mathbf{b}(k)$ соответственно; $\nabla_{\mathbf{a}_i(k)} \mathbf{E}(k), \nabla_{\mathbf{b}_i(k)} \mathbf{E}(k)$ – градиент критерия обучения $\mathbf{E}(k)$.

Для определения градиента критерия (4) введем апостериорную ошибку

$$\tilde{\mathbf{e}}(k+1) = (\mathbf{y}(k+1) - \mathbf{w}(k+1)\tilde{\mathbf{y}}(k+1))^T \mathbf{w}_i(k+1) \quad (6)$$

(здесь $\mathbf{w}_i(k+1)$ – i -я строка матрицы весов $\mathbf{w}(k+1)$), которая с учетом (2) и (3) имеет вид:

$$\begin{aligned} \tilde{\mathbf{e}}(k+1) &= (\mathbf{y}(k+1) - \mathbf{w}(k+1)\tilde{\mathbf{y}}(k+1))^T \mathbf{w}_i(k+1) = \\ &= (\mathbf{y}(k+1) - \mathbf{w}(k+1) \cdot [\varphi_i(\mathbf{a}(k+1)\mathbf{x}(k+1)) \circ \psi_i(\mathbf{b}(k+1)\mathbf{x}(k+1))])^T \mathbf{w}_i(k+1) = \\ &= (\mathbf{y}(k+1) - \mathbf{w}(k+1) \cdot [\varphi_i(\mathbf{v}(k+1)) \circ \psi_i(\mathbf{u}(k+1))])^T \mathbf{w}_i(k+1). \end{aligned} \quad (7)$$

Дифференцирование критерия $\mathbf{E}(k)$ по переменным a_{ij} и b_{ij} с использованием апостериорной ошибки в форме (7) дает следующий результат:

$$\begin{aligned} \frac{\partial \mathbf{E}(k)}{\partial a_{ij}} &= -\tilde{\mathbf{e}}^T(k) \frac{\partial \tilde{\mathbf{e}}(k)}{\partial a_{ij}} = -\tilde{\mathbf{e}}^T(k) \mathbf{w}_i(k) \psi_i(\mathbf{u}_i(k)) \frac{\partial \varphi_i(\mathbf{v}_i(k))}{\partial v_i} \mathbf{x}_j(k) = \\ &= -\tilde{\mathbf{e}}^T(k) \mathbf{w}_i(k) \psi_i(\mathbf{b}_i^T(k)\mathbf{x}(k)) \frac{\partial \varphi_i(\mathbf{a}_i^T(k)\mathbf{x}(k))}{\partial v_i} \mathbf{x}_j(k); \\ \frac{\partial \mathbf{E}(k)}{\partial b_{ij}} &= -\tilde{\mathbf{e}}^T(k) \frac{\partial \tilde{\mathbf{e}}(k)}{\partial b_{ij}} = -\tilde{\mathbf{e}}^T(k) \mathbf{w}_i(k) \varphi_i(\mathbf{v}_i(k)) \frac{\partial \psi_i(\mathbf{u}_i(k))}{\partial u_i} \mathbf{x}_j(k) = \\ &= -\tilde{\mathbf{e}}^T(k) \mathbf{w}_i(k) \varphi_i(\mathbf{a}_i^T(k)\mathbf{x}(k)) \frac{\partial \psi_i(\mathbf{b}_i^T(k)\mathbf{x}(k))}{\partial u_i} \mathbf{x}_j(k), \end{aligned} \quad (8)$$

где $\tilde{\mathbf{e}}(k) = (\tilde{e}_1(k), \tilde{e}_2(k), \dots, \tilde{e}_m(k))^T$ – вектор апостериорных ошибок обучения нейронов выходного слоя.

Теперь можно окончательно записать процедуры обучения векторов параметров $\mathbf{a}_i(k)$ и $\mathbf{b}_i(k)$ в виде:

$$\begin{cases} \mathbf{a}_i(k+1) = \mathbf{a}_i(k) + \eta_a(k) (\mathbf{y}(k+1) - \mathbf{w}(k+1)\tilde{\mathbf{y}}(k+1))^T \cdot \\ \cdot \mathbf{w}_i(k) \psi_i(\mathbf{b}_i^T(k)\mathbf{x}(k+1)) \frac{\partial \varphi_i(\mathbf{a}_i^T(k)\mathbf{x}(k+1))}{\partial v_i} \mathbf{x}(k+1), \\ \mathbf{b}_i(k+1) = \mathbf{b}_i(k) + \eta_b(k) (\mathbf{y}(k+1) - \mathbf{w}(k+1)\tilde{\mathbf{y}}(k+1))^T \cdot \\ \cdot \mathbf{w}_i(k) \varphi_i(\mathbf{a}_i^T(k)\mathbf{x}(k+1)) \frac{\partial \psi_i(\mathbf{b}_i^T(k)\mathbf{x}(k+1))}{\partial u_i} \mathbf{x}(k+1). \end{cases} \quad (9)$$

3. Имитационный эксперимент

Для исследования аппроксимирующих свойств предложенных архитектуры и алгоритма обучения проведен эксперимент по восстановлению двухмерной тестовой последовательности, имеющей хаотическую структуру, которая описывается уравнениями Манделброта [12]:

$$\begin{cases} x(k+1) = x^2(k) - y^2(k) + \theta_x, \\ y(k+1) = 2x(k)y(k) + \theta_y. \end{cases} \quad (10)$$

Аппроксимация проведена для 100 точек, обучение повторяется в ходе 40 эпох. Для алгоритма (5), (9) параметр шага составляет $h=0,01$; количество нейронов в скрытом слое равно 20. Для сравнения модель (10) была также аппроксимирована многослойным персептроном, который содержит два скрытых слоя по 7 нейронов в каждом. Количество точек, эпох обучения и параметр шага для MLP выбраны такими же, как и для сигма-пи сети. Результаты аппроксимационной процедуры приведены на рис. 2.

Точность аппроксимации оценивалась посредством вычисления среднеквадратичного отклонения. Значения отклонения для разных сетей и для каждого сигнала в отдельности приведены в таблице.

	Многослойный персептрон		Сигма-пи сеть	
	Сигнал $x(k)$	Сигнал $y(k)$	Сигнал $x(k)$	Сигнал $y(k)$
Средне-квадратичное отклонение	0.1030	0.1110	0.0210	0.0115

Полученный алгоритм обучения достаточно прост с вычислительной точки зрения и позволяет решать в реальном времени задачи эмуляции нелинейных нестационарных объектов управления. Особенно эффективным представляется применение данного алгоритма в задачах большой размерности.

Выводы

Научная новизна разработанной архитектуры искусственной нейронной сети заключается в удачном сочетании преимуществ радиально-базисных и сигмоидальных активационных функций. Градиентный алгоритм обучения позволяет настраивать синаптические веса сети в реальном времени с заданной точностью. Практическое значение имеют высокая скорость обучения и универсальные аппроксимирующие свойства предложенной сети, они будут особенно полезны при обработке многомерных функций векторного аргумента. В перспективе исследования предполагают разработку сигма-пи сети без использования процедуры прямого произведения входных векторов скрытого слоя.

Литература: 1. Narendra K.S., Parthasarathy K. Identification and control of dynamical systems using neural networks// IEEE Trans. on Neural Networks. 1990. 1, № 1. P. 4 – 27. 2. Cichocki A., Unbehauen R. Neural Networks for Optimization and Signal Processing. Stuttgart: Teubner, 1993. 526 p. 3. Elanaya S.V.T., Shin Yu. C. Radial basis function neural network for approximation and estimation of nonlinear stochastic dynamic systems // IEEE Trans. on Neural Networks. 1994. 5, № 4. P. 594–603. 4. Pham D.T., Liu X. Neural Networks for Identification, Prediction and Control. London: Springer-Verlag, 1995. 238 p. 5. Hornik K. Approximation capabilities of multilayer feedforward networks// Neural Networks. 1991. 4, P. 251 – 257. 6. Park J., Sandberg I.W. Universal approximation using radial-basis

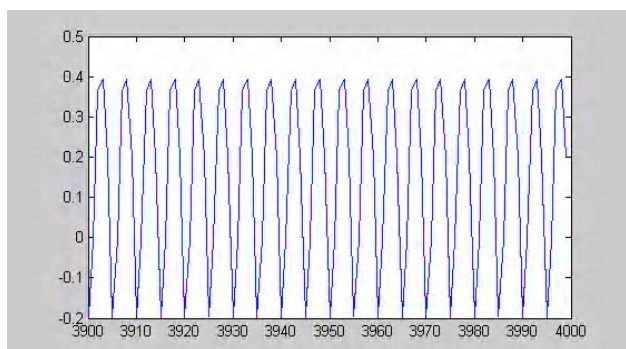
function networks// Neural Computation. 1991. 3. P. 246 – 257. 7. Luo Yu., Shen Sh. LP Approximation of Sigma-Pi neural networks//IEEE Trans. on Neural Networks. 2000. 11, № 6. P. 1485 - 1489. 8. Бодянский Е.В., Колодяжный В.В., Кулишова Н.Е. Прогнозирующая сигма-пи искусственная нейронная сеть // Матеріали міжнар. конф. з управління «Автоматика – 2001». Одеса, 2001. Т. 2. С. 141. 9. Бодянский Е.В., Кулишова Н.Е. Обобщенный алгоритм обучения прогнозирующей искусственной S-P нейронной сети// Проблемы бионики. 2001. Вып. 54. С. 10 – 14. 10. Бодянский Е.В., Кулишова Н.Е. S-P искусственная нейронная сеть на обобщенных формальных нейронах// Сб. науч. трудов 7-й междунар. конф. «Теория и техника передачи, приема и обработки информации». Харьков: ХТУРЭ, 2001. С. 404 – 405. 11. Bodyanskiy Y., Kolodyazhnyi V., Kulishova N. Generalized forecasting Sigma-Pi neural network//Intelligent Technologies – Theory and Applications. Ed. by Sincak P., Vascak J., Kvasnicka V., Pospichal J. Amsterdam. Berlin. Oxford. Tokyo. Washington: IOS Press. 2002. 36 p. 12. Mandelbrot B.B. Die fraktale Geometrie der Natur. Basel: Birkhaeuser Verlag, 1991. 491 S.

Поступила в редколлегию 12.12.2005

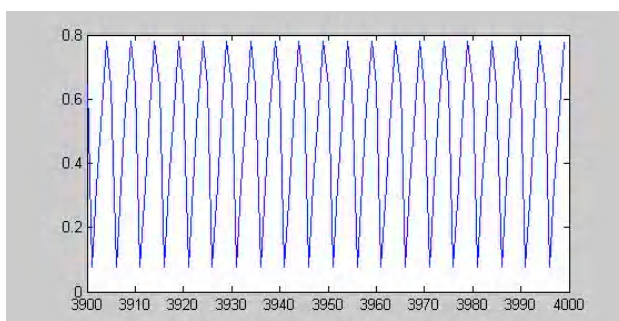
Рецензент: д-р техн. наук, проф. Алексеев О.П.

Бодянский Евгений Владимирович, д-р техн. наук, проф. кафедры искусственного интеллекта ХНУРЭ. Адрес: Украина, 61166, Харьков, пр. Ленина, 14, тел. (0572) 702-18-90.

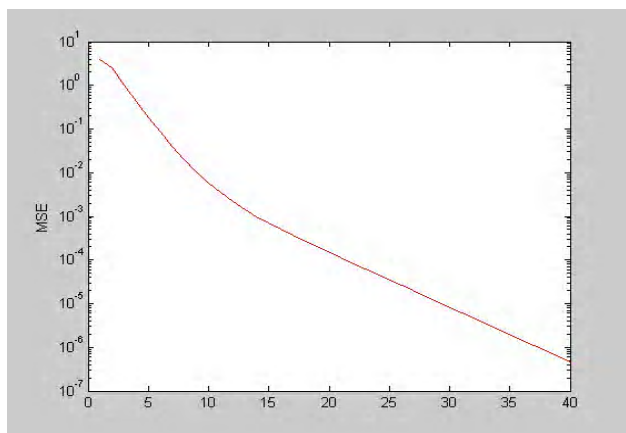
Кулишова Нонна Евгеньевна, канд. техн. наук, с.н.с., доц. каф. инженерной и компьютерной графики ХНУРЭ. Адрес: 61166, Харьков, пр. Ленина, 14, тел. (0572) 702-13-78.



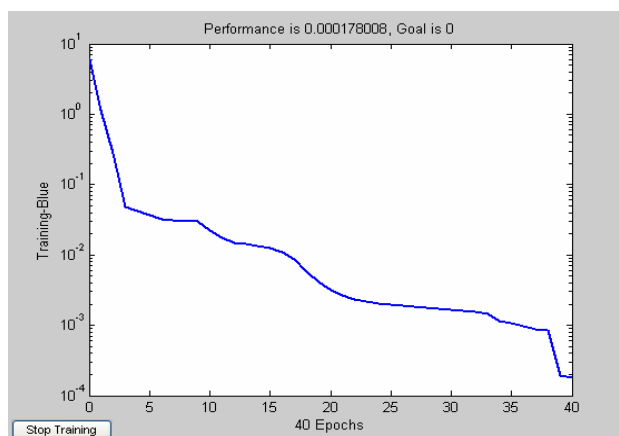
а



б



в



г

Рис. 2. Результаты имитационного эксперимента: а – аппроксимация $x(k)$; б – аппроксимация $y(k)$ (исходная функция практически совпадает с выходом сигма-пи сети); в – среднеквадратичная ошибка аппроксимации сигма-пи сети по эпохам (по оси абсцисс – логарифмические показатели); г – среднеквадратичная ошибка аппроксимации MLP по эпохам