

## НЕЙРОСЕТЕВАЯ АДАПТАЦИЯ ВХОДНОГО ВЕКТОРА КАК ЗАДАЧА НЕЛИНЕЙНОЙ ОПТИМИЗАЦИЯ

**Алёшин С.П.**

*к.т.н., доцент*

*Полтавского национального технического  
университета имени Юрия Кондратюка,  
Украина, г. Полтава*

**Бородина Е.А.**

*старший преподаватель*

*Полтавского национального технического  
университета имени Юрия Кондратюка,  
Украина, г. Полтава*

## НЕЙРОМЕРЕЖЕВА АДАПТАЦІЯ ВХІДНОГО ВЕКТОРУ ЯК ЗАДАЧА НЕЛІНІЙНОЇ ОПТИМІЗАЦІЇ

**Альошин С.П.**

*к.т.н., доцент*

*Полтавського національного технічного  
університету імені Юрія Кондратюка  
Україна, м. Полтава*

**Бородіна О.О.**

*старший викладач*

*Полтавського національного технічного  
університету імені Юрія Кондратюка  
Україна, м. Полтава*

## THE NEURAL NETWORK OF THE INPUT VECTOR ADAPTATION AS TASK OF THE NON-LINER OPTIMIZATION

**Aleshin S.P.**

*Ph.D. in Engineering Science, Associate Professor  
Poltava National Technical Yuri Kondratyuk University*

**Borodina E.A.**

*Senior lecturer*

*Poltava National Technical Yuri Kondratyuk University*

### **Аннотация**

Предложена методика адаптации входного вектора при решении задачи обратного распространения. Обучение обеспечивает построение адекватной модели исследуемого объекта, а градиентный спуск по входным сигналам позволяет продуктивно адаптировать множество входных факторов к требуемому классу состояния объекта исследования.

## **Анотація**

Запропоновано методику адаптації вхідного вектора при вирішенні задачі про-ратного розповсюдження. Навчання забезпечує побудову адекватної моделі досліджуваного об'єкта, а градієнтний спуск по вхідним сигналам дозволяє продуктивно адаптувати безліч вхідних факторів до необхідному класу стану об'єкта дослідження.

## **Abstract**

A method for adapting the input vector in solving the problem of the reverse distribution. Education enables the construction of an adequate model of the object, and the gradient descent allows for input signals is productive to adapt the set of input factors to a specific class state of the research object.

**Ключевые слова:** нейроуправление, адаптация, модификация, синаптическое пространство, инверсная динамика, признаки, классы.

**Ключові слова:** нейроуправління, адаптація, модифікація, синаптичний простір, інверсна динаміка, ознаки, класи.

**Keywords:** neural control, adaptation, modification, synaptic space, inverse dynamics, features classes.

**Постановка проблемы.** При принятии решений в управлении объектами и процессами в различных трудно формализуемых областях деятельности (экономика, образование, медицина, экология, геология и т.д.) возникает проблема продуктивного анализа априорных данных. Отмеченные объекты относятся к классу сложных систем [1], эффективное управление которыми, было и остается актуальной проблемой, требующей своего решения. Особый интерес в этом контексте вызывают сложные социотехнические системы (ССТС), как обобщенное отображение современных производственных предприятий, медицинских учреждений, учебных заведений и т.д. В них, к отмеченным выше особенностям, добавляется воздействие субъективного человеческого фактора, формализация и учет которого в автоматизированных системах принятия решений остается сегодня нерешенной проблемой.

Современный алгоритмический и программный инструментарий, позволяет оперативно конструировать нейросетевую среду, реализовать широкий класс архитектур нейросетей различной сложности и правил модификации синаптического пространства [2]. Вместе с тем, применение технологии искусственного интеллекта (ИИ) в управлении ССТС раскрывает ряд нерешенных теоретических и практических проблем, одной из которых является проблема адаптации входного вектора факторов к целевому состоянию объекта. Это требует создания продуктивных СППР, методов, технологий и программного обеспечения построения моделей, позволяющих автоматизировать базовые процессы при распознавании состояний, прогнозировании динамики поведения и выборе оптимальных управляю-

щих факторов исследуемых объектов на базе интеллектуальных технологий [3].

**Анализ последних публикаций.** Разработкой, теоретическим обоснованием и внедрением информационных технологий на основе искусственного интеллекта для эффективного управления занимались отечественные и зарубежные ученые, в частности: Глушков В.М., Т. Кохонен, С. Хайкин, Р. Хехт-Нильсен и др. В результате их деятельности созданы:

- основополагающие принципы и методы прикладной теории искусственного интеллекта в социальных и технических системах;
- обоснованные подходы к проектированию и моделированию социальных и технических систем;
- прикладная теория управления интеллектуальными производственными системами;
- концепция применения ситуационных центров как систем поддержки принятия решений для объектов повышенного риска: оборона, космос, ядерная энергетика;
- методы и алгоритмы многоуровневой дифференциальной диагностики в различных предметных областях;
- алгоритмические и программные продукты для оперативного многофакторного анализа данных в задачах классификации и прогноза;
- комплексная квалиметрия базовых процессов сложных систем.

Вместе с тем, применение технологии ИИ в управлении ССТС раскрывает ряд нерешенных теоретических и практических проблем, одной из которых является адаптация входного вектора факторов к целевому состоянию объекта [12, 13]. Это требует создания продуктивных СППР, методов, технологий и программного обеспечения построения моделей, позволяющих автоматизировать базовые процессы при распознавании состояний, прогнозировании динамики поведения и выборе оптимальных управляющих факторов исследуемых объектов на базе интеллектуальных технологий [14, 15].

**Цель:** Представление разработанной технологии построения продуктивного ансамбля нейросетевых моделей нахождения такого набора входных факторов, который оптимально надежно и быстро позволяет привести ССТС в целевое состояние. В качестве инструмента моделирования выbran нейроэмулятор среды StatSoft.

**Постановка задачи.** С точки зрения формализации процесса распознавания объекта необходимо построить отображение входного вектора данных на результат экспертизы, аппроксимирующее зависимость “поведение - вывод”. В общем случае признаки поведения и число классов объектов являются векторными величинами:

$$|Y(t)| = F |X(t)|, \quad (1)$$

где  $Y(t)$  – номер класса состояния объекта;  $X(t)$  – вектор текущих значений входных признаков;  $F$  – функционал преобразования массива

признаков в номер класса. Задача адаптации входного вектора к требуемому состоянию ССТС состоит в нахождении оптимального набора входного вектора факторов  $X_0(t)$ , соответствующего вектору целевого состояния  $Y_0(t)$  объекта исследования.

Этот процесс может быть формально представлен последовательностью процедур:

$$F : X(t) \rightarrow Y(t) \Leftrightarrow Y_0(t) \rightarrow F_0 : Y_0(t) \rightarrow X_0(t), \quad (2)$$

где  $Y_0(t)$  – целевой вектор состояния объекта;  $X_0(t)$  – вектор адаптированных значений входных признаков;  $F_0$  – продуктивный функционал модификации массива текущих входных факторов к искомому набору, адекватному целевому состоянию.

Реализация алгоритма (2) позволяет найти оптимальный набор входных факторов как последовательное решение прямой и обратной задачи распознавания состояний ССТС. Решение обратной задачи в нейросетевом формате обеспечивает нахождение таких значений сигналов на входе обученной сети, при которых достигается выходное состояние, соответствующее нужному классу [13]. В основу классификации предлагается положить использование обучаемой искусственной нейронной сети с обратным распространением ошибки как инструмента автоматического распознавания состояния.

### **Адаптация входных факторов к целевым состояниям объекта.**

Если актуальное состояние объекта не совпадает с целевым состоянием, то необходимо найти соответствующий набор входных факторов и привести его в это состояние. Для решения этой задачи следует найти функциональную зависимость состояний от входных факторов.

При этом минимизация целевой функции в постановке задачи выполняется итерационно в процессе обучения сети по алгоритму обратного распространения ошибки (или иным его модификациям) [14] применительно к входным данным исследуемого объекта:

$$\frac{1}{mn} \sum_{j=1}^n \sum_{i=1}^m (y_{ij} - d_{ij})^2 \Rightarrow (\mathfrak{R} \leq \mathfrak{R}_0), \quad (3)$$

где  $y_{ij}$  – вектор индикаторов выходных состояний;  $d_{ij}$  – результат обучения сети на  $j$  – выходе, при  $i$  - м примере обучающей выборки;  $j = 1, \dots, n$  – номер выхода сети;  $j = 1, \dots, m$  – номер примера;  $m, n$  – размерность массива примеров и числа выходных элементов сети;  $\mathfrak{R}_0$  – допустимые ограничения по условию задачи. В нейросетевом формате обучение заканчивается формированием ансамбля нейросетевых моделей. Анализ профилей результатов моделирования позволил выделить сети с приемлемыми для практики производительностью и ошибками на обучающих, контрольных и тестовых множествах. Установленная регрессионная зависимость определяет связь входных факторов и выходных индикаторов системы через мас-

сивы синаптических коэффициентов обученных нейросетевых моделей. Вариации входными факторами позволяют получать различные значения функции, что обеспечивает системного менеджера данными для принятия решений. Однако, актуальное состояние объекта обычно не совпадает с требуемым. Тогда на основании гипотезы компактности не совпадают и признаковые пространства, соответствующие этим состояниям. Поэтому требуется такой набор признаков, который максимально сближал бы классы текущего и требуемого состояний ССТС. Это задача обратная распознаванию образов [12,16]. После формализации процедур оценки текущего состояния и функциональной зависимости “вход-выход”, решалась задача формализации выбора оптимального признакового пространства в границах установленных дисциплинирующих условий. При небольшом числе изменяемых факторов (3-4) эта задача решалась простым перебором. В случае большого числа факторов (десятки – сотни) целесообразно воспользоваться одним из существующих градиентных методов адаптации входного вектора к заданной функции на выходе [12]. В этом случае метод обратного распространения ошибки последовательно применяется в два этапа:

- для подстройки параметров нейронной сети при модификации её синаптического пространства;
- для подстройки входного вектора через градиент функции ошибки по входным сигналам сети [5,8,12].

Эта процедура положена в основу решения задачи обратной к процессу распознавания состояний ССТС.

Задавались индикаторы соответствующие требуемому классу состояния объекта и подавались на вход обученной сети. Имея заданный ответ и ответ выданный сетью, вычислялся градиент функции ошибки по входным сигналам сети по известной технологии метода нагруженных двойственных сетей [9]. В соответствии со значениями элементов градиента изменялись значения входных сигналов сети в направлении снижения ошибки, что позволяет итерационно получить вектор входных сигналов, порождающих требуемый ответ. При этом синаптическая карта сети остается неизменной.

Таким образом, значения входных факторов по окончании процедуры подстройки являются результатом принятия решения нейронной сетью по переводу системы из одного состояния (класса) в другое. При этом необходимо выполнить анализ целевых критериев управления, исследовать возможные виды целевых функционалов, учесть условия и ограничения их применения.

Задача обучения НС имеет две фазы, первая представляет усвоение начальных функций НЭ и НК, которые приближенно соответствуют динамике управляемого О, а вторая – в поддержке адаптивного управления.

В первой фазе нейроэмулятор должен приобрести свойства инверсной динамики объекта, чтобы в первом приближении отображать динамику поведения ССТС или отдельных её состояний. Для этого вначале проводят

обучение нейронной сети (нейроэмулятора) в автономном режиме при многократной коррекции синаптического пространства. В процессе обучения при необходимости применяется обогащение выборки, выбор наиболее информативных признаков (редукция), выбор оптимальной архитектуры и метода обучения. Как результат сеть выдает отклик требуемой точности или подлежит модификации (увеличивается число задержанных сигналов добавлением соответствующего числа входов сети). Окончанием процесса является достижение адекватности модели в заданных границах установленных показателей.

Затем необходимо провести автономное обучение нейроконтроллера. Для уменьшения риска переобучения нейроэмулятора коэффициенты скорости НЭ уменьшают несколько раз в сравнении с соответствующими коэффициентами для нейроконтроллера.

После достижения необходимой точности автономной реакции НК проводится дополнительное обучение обеих нейросетей на примерах управления ССТС при различных значениях уставки.

Вторая фаза обучения нейросетей системы нейроруавления проходит в рабочем режиме управляемого объекта, что требует соблюдения строгого графика выполнения операций управления и коррекции параметров сетей. Главным является темп операций управления объектом, что обуславливает поступление реализаций  $Y(t)$ ,  $U(t)$   $Z(t)$ . Каждая новая реализация пополняет обучающую выборку для оперативного обучения. Чтобы избежать потери полезной информации, что содержится в этой реализации необходимо в течении одного такта выполнить достаточное количество эпох сетей. Число эпох можно уменьшить, увеличив коэффициент скорости (при этом увеличивается риск потери стабильности обучения).

Главной особенностью адаптивного управления ССТС является необходимость анализа данных и ППР в темпе управляемого базового процесса. Применение ИНС в системах АСУ требует согласования итерационных процессов обучения с требованиями реального времени. Как правило, ищется компромисс между требованиями РВ и эффективностью управления.

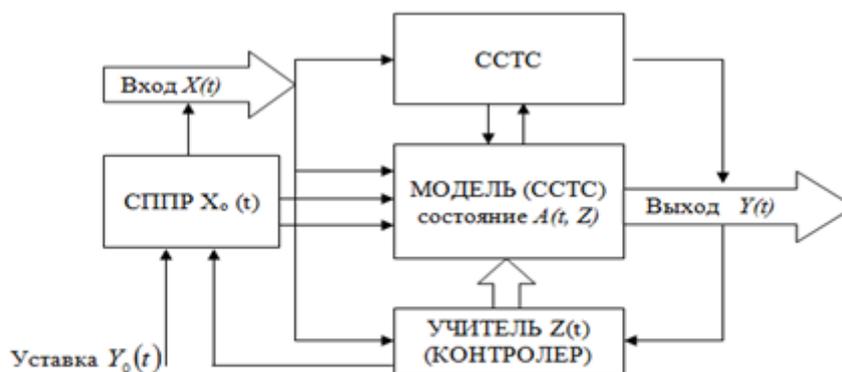


Рис.1. Структура процесса адаптации входного вектора

Состояние  $O$  отслеживается контроллером, определяющим правильность его реакции на управляющие факторы, задавая управление  $Z$ .

В основе решения задачи лежит приведение ошибки обучения нейронной сети к входному слою нейронов и расчету частных производных градиента функции ошибки по параметрам нейросети и входным факторам. Так как одновременно эти два процесса решаться не могут, то они разнесены во времени: сначала сеть обучается на примерах и приобретает свойства объекта исследования, затем вектор входных факторов модифицируется к этому состоянию. Это один цикл принятия решения о значении набора входных факторов в дискретном времени. Затем повторяется следующий цикл и процесс характеризуется уже как дискретно-непрерывный. Если время одного цикла удастся корректно связать с реальными процессами исследуемого объекта, учитывая особенности предметной области, то обновление расчетных данных позволяет обеспечить режим реального времени [16]. Градиентный спуск приводит к одному из возможных решений обратной задачи, что является исходным входным массивом для инструментальной поддержки решений ЛПР. Реализация такого подхода осуществляется с применением построения нейросетевой модели и её обучения по методу нагруженной сети [9]. Метод позволяет вычислить производных выходной функции невязки текущего и требуемого состояний ССТС по каждому элементу вектора входных факторов и запомнить их. При обратном функционировании эти ранее вычисленные производные, участвуют для вычисления градиентов по входным сигналам сети.

Таким образом, необходимо построить алгоритмы трех относительно самостоятельных процедур:

- построение адекватной модели текущего состояния исследуемого процесса ССТС;
- приближение вектора индикаторов текущего состояния ССТС к соответствующему вектору целевого состояния с заданной точностью;
- нахождение вектора входных факторов, соответствующего целевому состоянию в рамках допустимых отклонений.

Эти задачи решаются в комплексной схеме нейроуправления при нагруженной нейросети с оценками и сохранением в памяти на один такт производных выходных функций каждого элемента. Технологию инструментальной реализации этих задач рассмотрим с применением известных способах нейроуправления [13, 14, 15, 16].

Адаптивная система контроллера решает задачу сближения текущего значения выхода с требуемым, заданным вектором уставки. Это целесообразно применить к построению систем поддержки решений управления ССТС. В схемах нейроуправления функции учителя выполняет контроллер, оценивающий невязку текущего и целевого состояния объекта исследования и определяющий сигнал управления  $Z(t)$ . Обучение – согласно ОРР, при этом ошибка сети  $E^*$  определяется разностью уставки и текущего значения выхода эмулятора.

На рис.2 представлена одна из схем нейроуправления объектом. Нейроэмулятор выполняет функции адаптивной модели управляемого объекта. На его входы поступают текущие и задержанные во времени значения векторов управления  $Z(t), Z(t - \theta), Z(t - 2\theta)$ , и значения разности между векторами входа и уставки  $Y(t) - U(t), Y(t - \theta) - U(t - \theta)$ . Здесь  $\theta$  – величина шага задержки. Выходом нейроэмулятора является ожидаемое значение выхода управляемого объекта  $Y^*$ , а также значение вектора ошибки  $E$  для обучения нейроконтроллера.

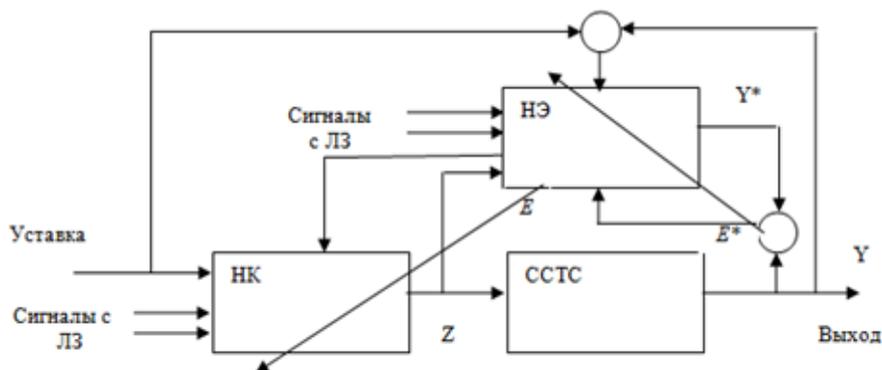


Рис. 2. Структурная схема адаптации входа нероуправления

Величина  $E^*(t) = Y(t) - Y^*(t)$  представляет вектор ошибки, который используется в алгоритме обучения нейроэмулятора.

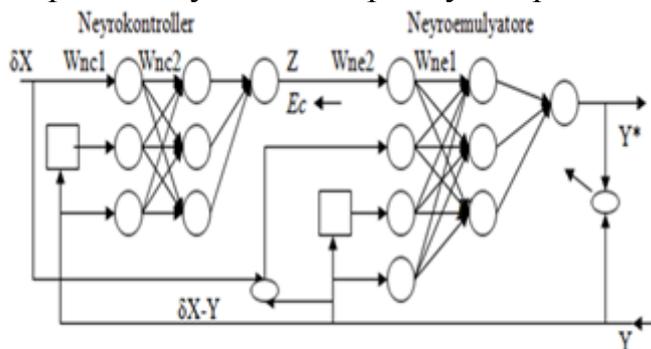


Рис.3. Каскадная схема нейроуправления

Наиболее удобно при этом использовать в нейроэмуляторе и контроллере схемы многослойного перцептрона с алгоритмом обратного распространения ошибки. Ошибка, поступающая в схему контроллера, представляется как ошибка, на выходе сети приведенная к первому слою в схеме нейроэмулятора. Это значит, что последовательность обучения двух сетей должна быть строго синхронизирована и взаимосвязана. Рассмотрим один из вариантов каскадной схемы сети и процессы формирования управляющих величин для принятия решений [5, 13]. На вход НЭ поступают:

$$E^*(t) = Y(t) - Y^*(t)$$

$Y(t)$  – текущие значения процесса на выходе;

$Y(t - \theta)$  – текущие значения процесса, задержанные на один такт  $\theta$ ;

$U(t) - Y(t)$  – текущее значение невязки на выходе системы;

$Z(t)$  – текущее управляющее воздействие.

Выходом нейроэмулятора являются значения выхода модели ССТС  $Y^*(t)$ .

При обучении НЭ осуществляется модификация синаптического пространства путем подстройки весовых коэффициентов в направлении антиградиента невязки:

$$w(t+1) = w(t) - \mu \left( \frac{\partial E}{\partial w} \right), \quad (4)$$

где  $\mu$  – коэффициент обучения;  $E$  – функция невязки (ошибка).

Обозначим ошибку НЭ как:

$$E^* = \frac{1}{2} (e^*)^2 = \frac{1}{2} (Y - Y^*) \quad (5)$$

$Y$  – представляет выход ССТС, а  $Y^*$  – реакцию НЭ:

$$Y^* = f_{out}^e \left\{ \sum_{i=1}^{N_2} w_i^{e_2} f^{e_2} \left[ \sum_{j=1}^{N_1} w_{ij}^{e_1} f^{e_1}(y_j^{e_1}) - b_i^{e_2} \right] - b_{out}^e \right\}, \quad (6)$$

где:  $f_{out}^e(\cdot), f^{e_2}(\cdot), f^{e_1}(\cdot)$  – сигнальные (активационные) функции нейронов 2-го та 1-го слоев;  $w_i^{e_2}, w_{ij}^{e_1}$  – весовые коэффициенты нейронов на участках между нейронами 2-3-го и 1-2-го слоев;  $b_i^{e_2}, b_{out}^e$  – величины порогов нейронов выхода и скрытого слоя.

Для нейронов на входе используется линейная сигнальная функция, а для скрытого слоя – сигмоидная. Порог на выходе обычно равен нулю. Для оценки производной функции невязки выполняются действия по известной методике [5, 9]:

На выходе НЭ нейроэмулятора процесс формализуется выражением:

$$\frac{\partial E^*}{\partial w_i^{e_2}} = -e^* f^{e_2}(v_i - b_i^{e_2}), \quad (7)$$

где  $v_i = \sum_{j=1}^{N_1} w_{ij}^{e_1} y_j^{e_1}$ .

Для связей между входами и скрытым слоем производная ошибки по весам имеет вид:

$$\frac{\partial E^*}{\partial w_{ij}^{e_1}} = -e^* \left\{ \sum_{i=1}^{N_2} w_i^{e_2} \left[ \frac{\partial f^{e_2}(v_i)}{\partial v_i} \right] \left( \frac{\partial v_i}{\partial w_{ij}^{e_1}} \right) \right\} = -e_i^{e_1} y_j^{e_1}, \quad (8)$$

$$\text{где: } \frac{\partial v_i}{\partial w_{ij}^{e_1}} = y_j^{e_1}; e_i^{e_1} = e^* \sum_{i=1}^{N_2} w_i^{e_2} \left[ \frac{\partial f^{e_2}(v_i - b_i)}{\partial v_i} \right] \quad (9)$$

Величина  $e_i^{e_1}$  представляется ошибкой, приведенной к  $j$ -му входу нейроэмулятора. Для входа, на который поступает вектор управления  $Z$ , эта величина представляет ошибку реакции НК, которая используется для коррекции синапсов НК. Подставляя полученные значения градиентов не-

вязки в исходное выражение (4), получаем алгоритм модификации весовых коэффициентов НЭ:

$$w_i^{e_2}(t+1) = w_i^{e_2}(t) + \mu^{e_2} e^* f^{e_2}(v_i), \quad (10)$$

$$w_{ij}^{e_1}(t+1) = w_{ij}^{e_1}(t) + \mu^{e_1} e_j^{e_1}, \quad (11)$$

где величина  $e_j^{e_1}$  рассчитывается по формуле (9). По такому же алгоритму осуществляется и подстройка весовых коэффициентов НК с учетом принятых обозначений входов и выходов.

Моделирование градиентного спуска с оценкой частных производных как по параметрам сети так и по входным сигналам, представляющим пространство признаков текущего состояния объекта, показало устойчивую сходимость итерационного процесса обучения сети с приемлемыми показателями качества

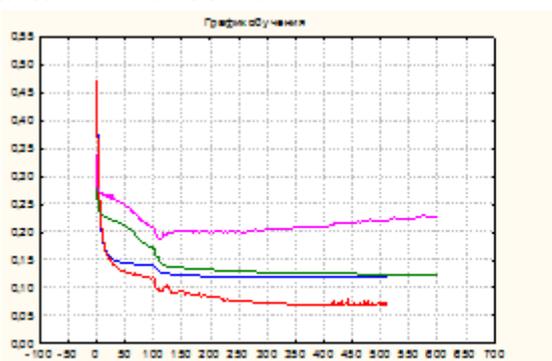


Рис. 4. Обучение моделей ВНС

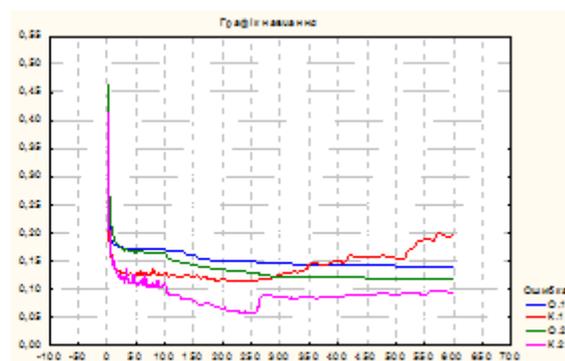


Рис. 5. Обучение моделей РБФ

Для различных условий моделирования получен ансамбль продуктивных нейросетей, которые могут использоваться в отдельных приложениях на основе их сохранения в формате основного кода.

### Выводы

1. Для автоматизации расчета факторов, приводящих актуальное состояние ССТС к целевому, необходимо найти функциональную зависимость её состояний от значений. Эта задача решена применением технологий нейроуправления и реализована моделями многослойных персептронов как обратная задача распознавания.

2. Расчет частных производных по входным факторам выходной функции невязки текущего и требуемого состояний ССТС и обобщенная оценка градиента выполнены на основе известных свойств двойственных нагруженных нейросетевых схем. Это позволило автоматизировать процесс определения элементов вектора градиента при реализации традиционного метода обратного распространения ошибки.

3. Практическая значимость результатов исследования состоит в создании алгоритмического и программного инструментария для перехода от автоматизированных к автоматическим системам адаптации пространства

входных признаков к пространству требуемых классов в системах распознавания образов с высокой степенью априорной неопределенности.

### Список использованной литературы

1. Полляк Ю.Г. Вероятностное моделирование на электронных вычислительных машинах. М.: Сов. Радио, 1971г.
2. Леонов А.Н., Васенев В.Н. и др. Моделирование в радиолокации. – М.:Сов.радио,1979 – 264с.
3. Бусленко Н.П. Моделирование сложных систем. 2-е изд. – М.: Наука, 1978. – 400с.
4. Хайкин, С. Нейронные сети: Полный курс. 2-е издание / Хайкин С.. М.: "Вильямс", 2006. – 1104 с.
5. Галушкин А.И. Нейрокомпьютеры и их применение на рубеже тысячелетий в Китае. Т.1 и 2 / А.И. Галушкин. М., 2004.- 367+464 с.
6. Вентцель Е.С. Теория вероятностей. Учебник для вузов. – М.: Наука, 1969. – 576с.
7. Лемешко Б.Ю., Миркин Е.П. Исследование критериев проверки гипотез, используемых в задачах управления качеством // Материалы VII международной конференции “Актуальные проблемы электронного приборостроения” АПЭП-2004. Новосибирск, 2004. – Т. 6. – С. 269 – 272.
8. Орлов А.И. О проверке однородности двух независимых выборок.// Заводская лаборатория. – 2003. – .69. №.1. – С.55-60//
9. Арнольд В.И. // Докл. АН СССР, том 114, N 4, 1957.
10. Hecht-Nielsen R. Kolmogorov's Mapping Neural Network Existence Theorem // IEEE First Annual Int. Conf. on Neural Networks, San Diego, 1987, Vol. 3, pp. 11–13.
11. Кобзарь А. И. Прикладная математическая статистика. Справочник для инженеров и научных работников. – М.: Физматлит, 2006. – 816 с.
12. Алешин С.П. Технологии построения модели менеджмента качества образовательного процесса в ВУЗе. – Международная научно-методическая конференция «Высшее образование в контексте Болонского процесса». Сб. науч. мат. – Полтавский национальный технический университет – Полтава.: ПНТУ,2008 . – с.68 – 71.