

Тархов Дмитрий Альбертович

**МАТЕМАТИЧЕСКОЕ МОДЕЛИРОВАНИЕ ФИЗИКО-ТЕХНИЧЕСКИХ  
ОБЪЕКТОВ НА ОСНОВЕ СТРУКТУРНОЙ И ПАРАМЕТРИЧЕСКОЙ  
АДАПТАЦИИ ИСКУССТВЕННЫХ НЕЙРОННЫХ СЕТЕЙ**

Специальность 05.13.18 – «Математическое моделирование, численные методы и комплексы программ»

**АВТОРЕФЕРАТ**

Диссертации на соискание учёной степени

Доктора технических наук

С-Петербург

2006 г.

Работа выполнена на кафедре «Высшая математика» Санкт-Петербургского государственного политехнического университета

Научный консультант

Доктор технических наук, профессор  
Малыхина Галина Фёдоровна

Официальные оппоненты: Нечаев Юрий Иванович, действительный член Академии Естественных Наук Российской Федерации, доктор технических наук, профессор, заслуженный деятель науки Российской Федерации;  
Шкодырев Вячеслав Петрович, доктор технических наук, профессор;  
Дорогов Александр Юрьевич, доктор технических наук, профессор

Ведущая организация: ЗАО «Котлин – Новатор»

Защита состоится 20 апреля 2006 г. в 14 часов на заседании диссертационного совета Д212.229.10 при Санкт-Петербургском государственном политехническом университете по адресу - 195251, Санкт-Петербург, Политехническая ул., 21 - СПбГПУ.

С диссертацией можно ознакомиться в фундаментальной библиотеке СПбГПУ.

Автореферат разослан

Учёный секретарь  
диссертационного совета  
д.т.н., профессор

Г.Ф. Малыхина

## 1. ОБЩАЯ ХАРАКТЕРИСТИКА ДИССЕРТАЦИИ

**Актуальность темы.** В настоящее время нейросетевая технология является одной из наиболее динамично развивающихся областей искусственного интеллекта. Она успешно применяется в различных областях науки и техники, таких как распознавание образов, системы диагностики сложных технических объектов (например, двигателя), экология и науки об окружающей среде (прогнозы погоды и различных катаклизмов), построение математических моделей, описывающих климатические характеристики, биомедицинские приложения (диагностика различных заболеваний, неинвазивное измерение уровня сахара в крови, идентификация личности), системы управления, геология (прогнозирование наличия полезных ископаемых) и т.д. На данный момент как в России, так и за рубежом, накоплен богатый опыт применения отдельных типов нейронных сетей к различным задачам. Созрела необходимость создания единой методологии разработки алгоритмов конструирования и обучения различного вида нейронных сетей применительно к решению широкого класса задач моделирования физико-технических объектов. Такая методология должна:

- Позволять исследователю сконструировать с помощью известных принципов или выбрать из имеющихся метод и алгоритм решения практически любой реальной задачи математического моделирования с помощью нейронных сетей, если другие подходы к задачам такого рода неизвестны или известны, но не устраивают по тем или иным причинам. Особенно актуальной, в частности, является задача построения простой модели сложного объекта по малой выборке с последующим уточнением модели по мере накопления новых данных в процессе функционирования объекта.

- Продемонстрировать свою работоспособность на достаточно широком классе практически интересных задач.

- Предоставить общие подходы к изучению условий сходимости конструируемых алгоритмов и к ускорению такой сходимости и повышению устойчивости соответствующих процессов.

В настоящее время такой методологии не существует, хотя многие глубокие исследования отдельных задач создают хорошие предпосылки для её возникновения.

В последние годы появился интерес к применениям нейронных сетей к частному виду таких задач - задачам математической физики. Это вызвано трудностями решения классическими методами многих практических задач (нелинейность моделей, сложность геометрии и т.д.). Неточность в задании параметров системы, начальных и краевых условий приводит к необходимости построения модели, которая слабо меняется при небольшом изменении этих данных. Нейросетевые модели обычно обладают такой устойчивостью. Недостатком существующих работ в данной области является то, что уравнение удовлетворяется в фиксированном наборе точек, а поведение между этими точками не учитывается. Большинство работ сводит поиск решения в классе нейронных сетей

к подбору линейно входящих параметров, что нивелирует достоинства нейросетевого подхода. Ещё одним существенным недостатком является отсутствие развитых алгоритмов подбора структуры сети, что вынуждает исследователей выбирать эту структуру априори, не опираясь на информацию об особенностях приближённого решения, проявляющихся в процессе вычислений.

Следующий круг вопросов связан с исследованием сходимости разработанных алгоритмов обучения. Определение условий и скорости сходимости стандартных алгоритмов обучения нейронных сетей в литературе практически не обсуждается, хотя соответствующие оценки сразу следуют из общих результатов для рассматриваемых алгоритмов. Анализ условий сходимости разработанных в диссертации алгоритмов существенно более сложен. Первая сложность заключается в изменении размерности пространства при итерациях, т.е. приходится рассматривать итерационный процесс в последовательности пространств, вложенных друг в друга. Вторая сложность связана с сочетанием шагов методов второго порядка и методов первого порядка, причём рассмотрение переменных функционалов может привести к замене неравенства Липшица оценкой Гёльдера с переменным показателем. Третья сложность связана с возможной заменой обратного оператора приближённым обратным с оценкой, меняющейся от шага к шагу. Отсутствие общих теорем о сходимости итерационных процессов такого рода не позволяла исследовать алгоритмы, разработанные с помощью обсуждающейся в диссертации методики.

Более актуальная для практики проблема заключается в большой ресурсоёмкости алгоритмов обучения нейронных сетей и большом времени обучения соответственно. Для решения этой проблемы возможны два пути.

Во-первых, интерес представляют такие модификации алгоритмов глобальной оптимизации, которые работоспособны в пространстве, размерность которого составляет сотни и тысячи, в отличие от существующих алгоритмов, которые работоспособны до размерности 20-30.

Во-вторых, требуется так модифицировать алгоритмы обучения, чтобы они достаточно эффективно работали в распределённой среде (Интернет). Более того, желательно реализовать распределённые варианты алгоритмов в виде соответствующих программных продуктов.

Хотя исходной целью изучения работы нейронных сетей было моделирование работы человеческого мозга, наиболее употребительные виды нейронных сетей оказались не вполне подходящими для решения данной задачи. Для моделирования этих процессов применяются и другие типы сетей, например, в виде системы нескольких связанных осцилляторов, совершающих квазипериодические колебания. Такие модели обладают рядом недостатков. Во-первых, реальное количество нейронов составляет миллиарды, поэтому остаётся неизвестным, не создаёт ли переход к такой большой системе принципиально новых качественных особенностей поведения. Во-вторых, интересно изучить особенности поведения многоуровневых систем, когда один блок нейронов, сильно связанных между собой, связан с другим блоком нейронов существенно слабее.

С прикладной точки зрения изучение этих вопросов весьма актуально. Первый круг возможных приложений – расшифровка мысленных команд по слабым электромагнитным колебаниям мозга и создание на этой основе новых человеко-машинных интерфейсов. Второй – лечение различных заболеваний с помощью слабых электромагнитных колебаний с отслеживанием результатов воздействия. Третий – бионические приложения, т.е. создание ведущих себя подобно мозгу интеллектуальных колебательных систем. Для реализации подобных систем нужно построить соответствующую теорию колебаний систем с бесконечным числом степеней свободы.

**Цель работы.** Диссертация посвящена созданию целостной методологии математического моделирования физико-технических объектов с помощью нейронных сетей. Такая методология строится на основе сочетания структурной и параметрической адаптации.

Достижение этой цели связано с выполнением следующих этапов исследования:

1. Единообразное и удобное для применения разрабатываемых методов математическое описание широкого класса нейронных сетей известных архитектур и конструирование на этой основе сетей новых типов.

2. Создание методологии конструирования и обучения нейронных сетей, применимой к упомянутым выше нейронным сетям и различным задачам математического моделирования физико-технических объектов и построение на этой основе ряда новых алгоритмов. При этом модели должны допускать развитие в процессе сбора новой информации об объекте и методы развития моделей должны входить в методологию.

3. Проверка применимости созданной методологии и построенных с её помощью алгоритмов на широком круге задач поиска приближённых решений обыкновенных дифференциальных уравнений и дифференциальных уравнений с частными производными.

4. Создание теоретических основ для анализа условий сходимости разработанных алгоритмов обучения нейронных сетей и разработка методов ускорения и повышения устойчивости работы входящих в них процедур локальной и глобальной оптимизации, в том числе на основе создания распределённых вариантов таких алгоритмов.

5. Создание теоретических основ для анализа почти периодических колебаний бесконечномерных систем взаимосвязанных осцилляторов.

6. Создание нейроэмулятора, позволяющего изучать работу алгоритмов обучения нейронных сетей и применять их к практическим задачам.

**Методы исследования.** Основой для создания и исследования разработанных алгоритмов является функциональный анализ, теория дифференциальных уравнений, метод группового учёта аргументов (МГУА) и эволюционное моделирование.

### **Научная новизна.**

1. Создана не существовавшая ранее методология конструирования и обучения нейронных сетей в задачах математического моделирования физических явлений в технических объектах на основе сочетания структурной и параметрической адаптации.

Проанализированы варианты реализации каждого этапа разработанной методологии, как известные так и новые и особенности их выбора в зависимости от решаемой задачи моделирования. В качестве примеров на основе данной методологии разработано несколько десятков новых методов и алгоритмов.

2. Определены новые виды нейронных сетей - вложенные нейронные сети и модификация RBF-сетей, включающая функции метода конечных элементов с указанием областей их возможного применения и особенностей обучения.

3. На основе разработанных в диссертации общих принципов созданы новые методы приближённого решения классических и неклассических задач математической физики.

4. Впервые определены многослойные дифференциальные нейронные модели и аналогичные модели с частными производными, а также рассмотрены особенности их построения и использования.

5. Доказаны новые теоремы о сходимости итерационных алгоритмов, обобщающих метод Ньютона, которые применяются к обучению нейронных сетей.

6. Разработаны новые методы ускорения и повышения устойчивости работы входящих в упомянутые выше методы и алгоритмы процедур локальной и глобальной оптимизации, в том числе на основе реализации их в глобальных сетях (Интернет).

7. Впервые определены осцилляторные нейросетевые модели бесконечной размерности и разработаны методы их изучения. Введены определения и доказана теорема о сходимости метода последовательных замен, что позволяет получать теоремы об устойчивости почти периодических колебаний таких систем в качестве частных случаев.

### **Практическая значимость.**

Разработанная автором методология математического моделирования на основе структурной и параметрической адаптации искусственных нейронных сетей позволяет специалисту в предметной области без особых усилий по программированию построить и исследовать математическую модель интересующего его физического или технического объекта. В качестве конкретных приложений можно указать следующие выполненные работы:

1. Разработанные автором методы применения нейронных сетей к задачам математической физики проиллюстрированы на примере моделирования волнового поля и определения оптимальной формы поверочной камеры калибратора переменного давления.

2. Разработанные автором методы применены к исследованию процессов теплообмена в системе «сосуды-ткани».

3. Под руководством автора создан нейроэмулятор Essence на JAVA 2, реализующий разработанные автором алгоритмы и позволяющий решать практические задачи построения нейросетевых моделей.

4. С помощью нейроэмулятора Essence было проведено указанное ниже исследование климатических характеристик Западной Сибири.

#### **Внедрение результатов работы.**

С помощью разработанных в диссертации методов и алгоритмов в ГОУВПО «Тюменский государственный нефтегазовый университет» успешно проводились исследования условий эксплуатации (климатических характеристик) техники в регионе Западной Сибири, что позволило найти скрытые зависимости в разнородных зашумлённых данных большого объёма.

На базе нейроэмулятора Essence в ООО «ГазЭнергоСервис» было создано специализированное программное обеспечение для моделирования и исследования температурного режима в Западной Сибири, что позволило спроектировать экономически целесообразный набор средств предпускового прогрева двигателя для землеройной техники в данном регионе.

**Апробация работы.** Основные результаты работы докладывались на следующих научных форумах:

- Второй научно-технический семинар «Современные системы контроля и управления электрических станций и подстанций (АСУ ТП) на базе микропроцессорной техники» в 2001 году,
- Международная конференция «Датчики и системы» в 2002 году,
- Международная конференция по мягким вычислениям и измерениям – SCM'2003, Санкт-Петербург, СПбГЭТУ «ЛЭТИ»,
- VI Всероссийская научно-техническая конференция «Нейроинформатика-2004», Москва, МИФИ,
- 5-я международная научно-техническая конференция «Компьютерное моделирование 2004» СПб.,
- Международная конференция по мягким вычислениям и измерениям – SCM'2004, Санкт-Петербург, СПбГЭТУ «ЛЭТИ»,
- 10 международный симпозиум ИМЕКО «TC7 International Symposium on Advances of Measurement Science» 2004, Санкт-Петербург,
- Международная научно-техническая конференция «Искусственный интеллект. Интеллектуальные и многопроцессорные системы» 2004, Казивели, Крым,
- VII Всероссийская научно-техническая конференция «Нейроинформатика-2005», Москва, МИФИ,

- Международная научно-техническая конференция «Интеллектуальные и многопроцессорные системы» (ИМС-2005) и научные молодежные школы «Высокопроизводительные вычислительные системы» (ВПВС-2005) и «Нейроинформатика и системы ассоциативной памяти» (Нейро-2005),
- Санкт-Петербургский городской семинар по нейронным сетям 25 ноября 2005 года,
- Научный семинар кафедры «Высшая математика» СПбГПУ (дважды).

На международных научно-технических конференциях «Искусственный интеллект. Интеллектуальные и многопроцессорные системы - 2004» и «Интеллектуальные и многопроцессорные системы - 2005» доклады в числе лучших в секции «Нейронные сети и нейросетевые технологии» опубликованы в специальных выпусках журнала Известия ТРТУ.

Созданный под руководством автора с помощью нейроэмулятора Essence "Multi-Layer Perceptron" Java Bean на международном конкурсе IBA "Beans - 98" (конкурс проводился под эгидой IBM) занял первое место.

#### **На защиту выносятся:**

1. Методология применения нейронных сетей для решения задач математического моделирования физико-технических объектов на основе структурной и параметрической адаптации и созданные на её основе методы и алгоритмы.
2. Методы и алгоритмы решения задач математической физики, построенные на основе известных и новых видов нейронных сетей и разработанных автором методов определения их структуры и весов.
3. Теоремы о сходимости итерационных алгоритмов, обобщающих метод Ньютона и применяемых к анализу сходимости разработанных алгоритмов, и методы ускорения и повышения устойчивости работы входящих в эти алгоритмы процедур локальной и глобальной оптимизации, в том числе ориентированные на использование в глобальных сетях (Интернет).
4. Осцилляторные нейросетевые модели бесконечной размерности и методы их исследования. Теорема о сходимости метода последовательных замен, что позволяет получать теоремы об устойчивости почти периодических колебаний таких систем в качестве частных случаев
5. Созданный на основе теоретических разработок, изложенных в данной диссертации, и под руководством автора, пакет Essence.

**Структура диссертации.** Диссертация содержит введение, шесть глав, заключение и список литературы, содержащий 200 источников, изложена на 333 страницах, включая 86 рисунков.

## **2. СОДЕРЖАНИЕ ДИССЕРТАЦИИ**

**Во введении** обоснована актуальность диссертации, определён предмет и цели исследований, а также кратко описано содержание диссертации по главам.



**В первой главе** систематизированы основные модели нейронных сетей на основе единого формализма. В начале главы сформулированы требования к нейросетевым моделям и основные задачи моделирования, к решению которых в следующих главах приложены нейронные сети.

В простейшей ситуации перед построением математической модели некоторой системы накапливают набор экспериментальных данных. Далее по этим данным строят зависимость  $y = f(x)$ , которую далее используют для изучения тех или иных особенностей поведения системы. Конкретная процедура построения такой зависимости определяется видом функции  $f$  и функционалом ошибки. Сформулируем желательные требования к функции  $f$ :

1. Это должна быть сколь угодно сложная функция, построенная из простых элементов.
2. Выбор конкретной структуры  $f$  должен сводить задачу к подбору конечного множества параметров.
3. Процедура подбора параметров должна быть устойчива к ошибкам наблюдений и вычислительным погрешностям.

Некоторые известные математические модели такого рода в научной литературе принято называть искусственными нейронными сетями. В диссертации под нейронными сетями понимаются такие модели и их естественные модификации.

Таким образом, нейронная сеть определяется правилом построения (архитектура сети), количеством элементов и способом их соединения друг с другом (структура сети) и коэффициентами, которые определяют конкретную функцию  $f$  (веса сети).

В первом параграфе на примере линейной регрессии приводятся основные виды функционалов ошибки, включая критерии МГУА, и сформулированы подходы к подбору структуры искомой зависимости. В дальнейшем эти результаты будут перенесены на нейронные сети.

Во втором параграфе сформулирована задача построения нелинейной регрессии общего вида, частным случаем которой является большинство видов нейронных сетей, и анализируются особенности постановки и решения этой задачи в случае, когда выборка пополняется. Кроме того, приведены методы построения квазилинейной регрессии, которая получается в том случае, когда искомую зависимость представляют в виде  $y = \sum_{i=1}^m w_i f(\mathbf{b}_i, \mathbf{x})$ .

В третьем параграфе рассмотрены статические нейронные сети, т.е. нейросетевые модели, не содержащие в явном виде время. Начинается анализ сетей этого вида с многослойного персептрона (многослойной сети прямого распространения). Обозначим вектор входов  $l$ -го слоя  $\mathbf{y}_{l-1}$ , а вектор выходов  $\mathbf{x}_l$ . Тогда указанная нейронная сеть описывается рекуррентными соотношениями

$$\mathbf{y}_l = \mathbf{W}_l \mathbf{x}_l, \tag{1}$$

$$\mathbf{x}_l = \varphi_l(\mathbf{y}_{l-1}), \tag{2}$$

где  $\mathbf{x}_0 = \mathbf{x}$  - вход сети;  $\mathbf{y}_L = \mathbf{f}(\mathbf{x}, \mathbf{w})$  - выход,  $\mathbf{W}_l$  - матрица весов  $l$ -го слоя,  $\mathbf{w}$  - совокупность этих матриц,  $\varphi_l$  - активационная функция, которая действует по координатам. Обучение сети состоит в поиске вектора  $\mathbf{w}$ , для которого суммарная ошибка по всем примерам (опытам) минимальна, например

$$E(\mathbf{w}) = \sum_{n=1}^N e_n = \min, \quad (3)$$

$e_n$  - ошибка для  $n$  - одного примера.

Формулы (1), (2) позволяют легко вычислить градиент обычного функционала ошибки по весам сети с помощью формулы дифференцирования сложной функции и применить для обучения персептрона один из методов нелинейной оптимизации. Также легко можно вычислить производные второго порядка, что позволяет применить для обучения метод Ньютона, рассмотренный в общем виде в четвертой главе.

Далее введена полная сеть прямого распространения, которая отличается от многослойного персептрона тем, что формула (1) заменяется выражением

$$\mathbf{y}_l = \sum_{l_i=0}^l \mathbf{W}_{l_i, l} \mathbf{x}_{l_i}. \quad (4)$$

Недостатком упомянутых выше сетей является то, что мы не можем рассмотреть процедуры их модификации, работающие на уровне отдельных связей. Для решения этой проблемы предложено использовать нейронные сети прямого распространения с частичной структурой связей.

С каждой такой нейронной сетью связан потоковый граф, вершины которого соответствуют нейронам, а также входам и выходам сети, а дуги – связям. Пусть  $V$  - множество вершин сети, тогда множество дуг  $D$  является подмножеством  $V \times V$ , т.е. каждой дуге соответствует упорядоченная пара вершин  $(v_1, v_2)$  – из первой дуга исходит, а во вторую – входит. Каждой вершине  $v$  сопоставим активационную функцию  $\varphi_v$ , каждой дуге  $(u, v)$  – вес  $w_{u,v}$ . Сеть прямого распространения не имеет циклов, т.е. её вершины всегда можно пронумеровать так (т.е.  $V$  можно считать множеством натуральных чисел), что  $u < v$  для каждой дуги.

Обозначим  $y_v$  вход соответствующего вершине с номером  $v$  нейрона, а  $x_v$  - его выход. Тогда формулы (1), (2) для сетей прямого распространения с частичной структурой связей приобретают вид

$$y_v = \sum_{u:(u,v) \in D} w_{u,v} x_u \quad (5)$$

т.е. суммирование производится по всем входящим в нейрон дугам,

$$x_v = \varphi_v(y_v) \quad (6)$$

Далее в данном параграфе поставлена задача кластеризации и анализируются решающие эту задачу сети Кохонена. От описанных ранее сетей прямого распространения сети Кохонена

отличаются тем, что не требуют задания выходов сети на обучающем множестве, т.е. обучение проходит только по входам сети.

Сети Кохонена можно применить и к задаче построения нелинейной регрессии. Простейший способ сделать это – подавать на выход сети не номер выигравшего нейрона, а линейные комбинации выходов слоя Кохонена. Такая сеть называется сетью Гроссберга. Обычно сеть Гроссберга обучается намного быстрее, чем многослойный персептрон или иная сеть, которая используется для решения задачи аппроксимации. Её недостаток состоит в том, что для получения такой же точности требуется намного больше нейронов.

Если искать выход сети в виде

$$y = \sum_{j=1}^m w_j \varphi_j(\|\mathbf{x} - \mathbf{c}_j\|), \tag{7}$$

где  $\mathbf{c}_j$  - некоторые центры, вокруг которых происходит аппроксимация, тогда получаем сети радиальных базисных функций (RBF – сети). Они предпочтительны в задачах интерполяции, а персептрон лучше применять для экстраполяции и для моделирования задач со скачками.

В четвёртом параграфе первой главы рассмотрены динамические нейронные сети. Первым видом сетей такого типа является многослойный персептрон с временными задержками. Для такой сети формулы (1) и (2) заменятся на соотношения

$$\mathbf{y}_l(n) = \sum_{t=0}^{p_l} \mathbf{W}_l(t) \mathbf{x}_l(n-t), \tag{8}$$

$$\mathbf{x}_l(n) = \mathbf{u}_l(\mathbf{y}_{l-1}(n)). \tag{9}$$

Аналогично описываются полные сети с временными задержками, для которых на вход нейрона могут подаваться линейные комбинации выходов всех предыдущих слоёв, а не только непосредственно предшествующего.

Для рассмотрения временных сетей с частичной структурой связей удобно каждому временному отсчёту сопоставить свой экземпляр множества вершин  $V$ . Множество дуг  $D$  является подмножеством  $V \times V$ , при этом нейроны можно пронумеровать так, чтобы связи шли от нейрона с меньшим номером к нейрону с большим номером, т.е.  $u < v$  для каждой дуги. Дуге соответствует упорядоченная пара вершин  $u(n-t), v(n)$ ,  $t=0,1,2,\dots$ . Каждой вершине  $v(n)$  сопоставим активационную функцию  $\varphi_v$ , каждой дуге  $u(n-t), v(n)$  – вес  $w_{u,v}(t)$ . Обозначим  $y_v(n)$  вход соответствующего нейрона, а  $x_v(n)$  - его выход. Тогда формулы (5), (6) приобретают вид

$$y_v(n) = \sum_{u:(u,v) \in D} w_{u,v}(t) x_u(n-t), \tag{10}$$

т.е. суммирование производится по всем входящим в нейрон дугам,

$$x_v(n) = \varphi_v(y_v(n)). \tag{11}$$

Когда классифицируемая выборка меняется со временем, возникают различные варианты задач кластеризации.

- Первый вариант - кластеризация пополняющейся выборки. При этом без особых изменений можно применять любой последовательный алгоритм кластеризации, включая сети Кохонена.
- Второй вариант заключается в том, что в один кластер относятся точки близкие не только по пространственным координатам, но и по времени.
- Третий вариант возникает, когда одному моменту времени соответствует не один, а целый набор входных векторов и нас интересует динамика кластеров, т.е. изменение их центров, размеров и т.п. Самый простой подход к решению такой задачи – обучить сети Кохонена для каждого временного отсчёта. Достоинством такой процедуры является её простота и возможность распараллеливания, недостатком – сложность интерпретации результата, т.е. анализ динамики весов сетей. Далее можно попытаться спрогнозировать будущие кластеры. Наиболее простой способ это сделать – использовать последовательность весов  $\{\mathbf{w}_i(t)\}_{t=1}^n$  для того, чтобы спрогнозировать  $\mathbf{w}_i(n+1)$ . Более тонкий результат можно получить, если использовать не только веса сети Кохонена, но и размеры кластеров – среднеквадратичное отклонение от центра или набор главных компонент множества элементов кластера.

В четвёртом параграфе проанализированы рекуррентные сети, т.е. сети с обратными связями. В данной ситуации формулы (10) – (11) остаются такими же, но исчезает условие  $u < v$  с одним исключением. Если при  $t=0$  будет существовать как  $w_{u,v}(0) \neq 0$ , так и  $w_{v,u}(0) \neq 0$ , тогда мы сможем вычислить выход сети при известных входах только с помощью решения системы уравнений (10) – (11). Такая сеть неудобна для использования, поэтому будем предполагать, что нейроны можно пронумеровать таким образом, чтобы  $w_{u,v}(0) \neq 0$  только при  $u < v$ .

Далее в данном параграфе показано, что большинство видов нейронных сетей может быть описано следующими рекуррентными соотношениями. Обозначим  $\mathbf{x}(n)$  – вектор входов сети,  $\mathbf{z}(n)$  – вектор внутренних состояний, которые не являются ни входами, ни выходами,  $\mathbf{y}(n)$  – вектор выходов сети. Тогда искомые формулы можно записать в виде:

$$\mathbf{z}(n) = \sum_{t=0}^T \mathbf{W}_{zx}(t)\mathbf{x}(n-t) + \sum_{t=0}^T \mathbf{W}_{zz}(t)\mathbf{z}(n-t) + \sum_{t=0}^T \mathbf{W}_{zy}(t)\mathbf{y}(n-t) . \quad (12)$$

$$\mathbf{y}(n) = \sum_{t=0}^T \mathbf{W}_{yx}(t)\mathbf{x}(n-t) + \sum_{t=0}^T \mathbf{W}_{yz}(t)\mathbf{z}(n-t) . \quad (13)$$

Для большинства архитектур нейронных сетей отображение  $\mathbf{z}$  действует по координатам и определяется функцией активации соответствующего нейрона. Такие обозначения позволяют единообразно описать представленные выше виды нейронных сетей (это сделано в диссертации) и применить к нейронным сетям стандартные методы теории управления.

После этого в качестве частных случаев рассматривается несколько известных нейросетевых архитектур:

- Сети Хопфилда, для которой равенства (11) и (12) имеют вид

$$\mathbf{z}(n) = \mathbf{W}\mathbf{y}(n-1), \quad (14)$$

$$\mathbf{y}(n) = \mathbf{\Psi}(\mathbf{z}(n)). \quad (15)$$

Нейронные сети Хопфилда обычно применяют для решения задачи запоминания информации и для поиска экстремума некоторой функции. Например, таким образом можно сформулировать известную задачу коммивояжёра. Однако стандартное применение сетей Хопфилда в этих задачах не является очень эффективным, вероятно потому, что оно основано на методе градиентного спуска, который в сложных задачах сходится медленно.

- Сети Хемминга, которые являются некоторой модификацией сетей Хопфилда. Эта архитектура получается, если на входе сети Хопфилда добавить слой нейронов, которые вычисляют меру рассогласования входного вектора с набором эталонов.

• Двухнаправленная ассоциативная память (сеть Коско), которая является модификацией сети Хопфилда, предназначенной для установления связей между двумя векторами  $\mathbf{x}$  и  $\mathbf{y}$  и функционирует следующим образом:

$$\mathbf{z}_1(n) = \mathbf{W}\mathbf{x}(n-1); \mathbf{y}(n) = \mathbf{\Psi}(\mathbf{z}_1(n)); \mathbf{z}_2(n) = \mathbf{W}^T\mathbf{y}(n); \mathbf{x}(n) = \mathbf{\Psi}(\mathbf{z}_2(n)). \quad (16)$$

- Сети Джордана. Этот вид сетей получается из многослойного персептрона, если на его вход подать помимо входного вектора выходной с задержкой на один или несколько тактов. Тогда функционирование сети описывается соотношениями  $\mathbf{y}_0(n) = \mathbf{W}_0\mathbf{x}(n) + \mathbf{W}(1)\mathbf{y}(n-1)$ ,  $\mathbf{y}_l(n) = \mathbf{W}_l\varphi_l(\mathbf{y}_{l-1}(n))$ ,  $\mathbf{y}(n) = \mathbf{W}_L\varphi_L(\mathbf{y}_{L-1}(n))$ .

- Сети Элмана, так же как и сеть Джордана получается из персептрона введением обратных связей, только связи идут не от выхода сети, а от выходов внутренних нейронов. В классическом варианте сеть имеет один слой нейронов и функционирует в соответствии с формулами  $\mathbf{y}_0(n+1) = \mathbf{W}_0\mathbf{x}(n) + \mathbf{W}(1)\varphi(\mathbf{y}_0(n))$ ,  $\mathbf{y}(n) = \mathbf{W}_1\varphi(\mathbf{y}_0(n))$ .

В пятом параграфе первой главы поставлена задача построения робастной математической модели по разнородным данным, включающим как уравнения, так и экспериментальные наблюдения. При этом уравнения могут быть как обыкновенными дифференциальными, так и в частных производных, интегральными, интегро-дифференциальными и т.д. Построение модели может сводиться к решению краевой задачи, а может состоять в подборе уравнения и решения по экспериментальным данным. Для исследования упомянутых выше задач применяются различные аналитические и численные подходы (метод сеток, конечных элементов, граничные интегральные уравнения, асимптотические разложения и др.). Однако это не мешает рассматривать использование нейронных сетей в качестве новой и более перспективной (в силу своей универсальности)

методологии решения как старых, так и новых задач такого типа. Проверить работоспособность предлагаемых подходов нужно в процессе решения разнообразных практических задач.

В шестом параграфе описаны некоторые методы глобальной оптимизации. Обычно применяемые для обучения нейронных сетей алгоритмы являются локальными, т.е. позволяют найти только локальный минимум. Так как функция ошибки в большинстве случаев многоэкстремальная, локальные методы обычно не позволяют найти наилучшее решение исходной задачи. Для того, чтобы приблизиться к такому решению, требуется применить процедуру, позволяющую найти приближённый глобальный минимум. Трудность в изучении алгоритмов глобальной оптимизации состоит в том, что либо их сходимость не доказана, либо теоретические оценки показывают крайне низкую их эффективность в задачах большой размерности, т.е. таких, которые нас интересуют. Ставится вопрос о создании алгоритмов, более приспособленных к обучению нейронных сетей, т.е. работоспособных в ситуации, когда число подбираемых переменных (весов сети) составляет сотни и тысячи.

Наступление эпохи Интернет привело к тому, что многие методы и парадигмы вычислений оказались устаревшими, а на первый план выходят совсем другие. Одной из таких парадигм, актуальность которой со временем будет только возрастать, являются распределённые вычисления. Под распределёнными вычислениями здесь подразумеваются процессы решения одной задачи на нескольких (многих) компьютерах, сообщающихся между собой через Интернет. В данном параграфе анализируется специфика задачи более конкретно, применительно к нейронным сетям.

Предыдущие виды нейронных сетей не отражают в достаточной степени адекватно процессы, происходящие в мозге. Для математического моделирования этих процессов в работах по нейронным сетям применяются и другие типы сетей, например, в виде системы из нескольких связанных осцилляторов, совершающих квазипериодические колебания. Эта модель тоже не является идеальной, хотя бы потому, что мозг это целостная система, состоящая из миллиардов нейронов. Поэтому более адекватным представляется рассмотреть систему из бесконечного множества осцилляторов, связи между которыми ослабевают при переходе от одной группы к другой. При этом место квазипериодических колебаний занимают почти периодические колебания с бесконечным базисом частот. В седьмом параграфе первой главы ставится задача заложить основы такой теории.

Представленные выше методы могут привести к созданию распределённой по Интернет интеллектуальной системы, способной решать разнообразные задачи моделирования. Перспективы создания такой системы обсуждаются в восьмом параграфе первой главы. В качестве первого шага ставится задача создания нейроэмулятора на языке JAVA 2, реализующего созданные автором алгоритмы.

Для дальнейшего развития теории построения и обучения нейронных сетей в задачах математического моделирования физико-технических объектов необходимо решить следующие вопросы:

1. Можно ли на базе стандартных видов нейронных сетей образовать новые их виды, более подходящие для некоторых задач моделирования?
2. Как удачным образом подобрать начальные веса сети в связи с тем, что известна сильная зависимость процесса оптимизации от выбора начального приближения?
3. Как выбрать структуру сети или организовать процесс модификации структуры во время обучения в зависимости от решаемой задачи моделирования?

Ответам на эти вопросы посвящена **вторая глава** диссертации. Главная задача данной главы состоит в создании единой методологии определения структуры и весов нейронных сетей в процессе решения широкого круга задач математического моделирования. Отсутствие такой методологии является серьёзным препятствием, затрудняющим практическое использование нейронных сетей и заставляющим многих исследователей заново разрабатывать подходы к решению каждой конкретной задачи. В качестве примеров на основе данной методологии разработан ряд новых методов и алгоритмов:

- Генетический метод определения структуры многослойного персептрона и реализующий его алгоритм.
- Генетический метод определения структуры сети прямого распространения с частичной структурой связей и реализующий его алгоритм.
- Генетический метод определения структуры сети Кохонена и реализующий его алгоритм.
- Двойной генетический метод построения коллектива нейронных сетей и реализующий его алгоритм.
- Метод построения RBF – сети, основанный на кластеризации ошибок и главных компонентах и реализующий его алгоритм.
- Метод построения коллектива RBF-сетей и реализующий его алгоритм.
- Метод построения системы вложенных сетей и реализующий его алгоритм.
- Генетический метод определения структуры сети прямого распространения с временными задержками и частичной структурой связей и реализующий его алгоритм.
- Блочный генетический метод определения структуры сети прямого распространения с временными задержками и частичной структурой связей и реализующий его алгоритм.
- Метод обучения сети Кохонена в случае кластеров, следующих друг за другом и реализующий его алгоритм.
- Метод сопоставления нейронов сети Кохонена в разные моменты времени и реализующий его алгоритм.

- Генетический метод построения коллектива нейронных сетей прямого распространения с временными задержками и реализующий его алгоритм.
- Двойной генетический метод построения коллектива нейронных сетей прямого распространения с временными задержками и реализующий его алгоритм.
- Метод кластеризации ошибок для известных моментов событий и реализующий его алгоритм.
- Метод кластеризации ошибок для неизвестных моментов событий и реализующий его алгоритм.
- Метод построения коллектива динамических RBF-сетей и реализующий его алгоритм.
- Метод последовательной метод динамической кластеризации с прогнозом и реализующий его алгоритм.
- Генетический метод определения структуры сети с обратными связями по набору выборок и реализующий его алгоритм.
- Метод моделирования неизвестного объекта и реализующий его алгоритм.
- Генетический метод определения структуры сети с обратными связями по пополняемой выборке и реализующий его алгоритм.
- Генетический метод определения структуры сети Хопфилда и реализующий его алгоритм.
- Многорядный метод определения структуры сети Хемминга и реализующий его алгоритм.
- Многорядный метод определения структуры сети Коско и реализующий его алгоритм.
- Метод выращивания сети Джордана для моделирования неизвестного объекта и реализующий его алгоритм.

Разработаны особые способы применения данных алгоритмов к различным задачам математического моделирования.

В первом параграфе второй главы решается задача построения статической нейронной сети прямого распространения по статической выборке. Анализируются известные и новые подходы к определению начальных значений весов сети, метод главных компонент предварительной настройки многослойного персептрона, процедуры добавления и удаления нейрона и слоя сети. На основе этих процедур строится генетический алгоритм определения структуры многослойного персептрона. При этом для задания работы генетического алгоритма необходимо ввести основные операции – мутации, транслокации и скрещивание, а также определить критерии отбора сетей в следующее поколение. Варианты таких операций определяются в естественных для нейронных сетей терминах, и строятся алгоритмы построения нейронных сетей на их основе. Рассмотренная в третьем параграфе первой главы конструкция сети с частичной структурой связей позволяет удобным образом строить генетический алгоритм подбора её архитектуры.



Перспективным на начальном этапе построения нейронной сети является многорядный алгоритм определения структуры сети прямого распространения с частичной структурой связей. На  $l+1$ -м шаге его работы рассматриваем сети вида  $y_{l+1} = w_0 + w_i y_{i,l} + w_j y_{j,l} + w_{\varphi,k} \varphi(y_{k,l})$ . Здесь  $y_{i,l}$  -  $i$ -й выход сети, получившийся после  $l$ -го шага. В разных моделях можно выбрать различные функции активации  $\varphi$ . Параметры моделей подбираются с помощью какого-либо алгоритма оптимизации функционала ошибки. При этом для каждой выходной переменной в выборке на каждом ряду строится свой набор моделей. Если нет оснований считать выходы рассматриваемой модели связанными, тогда эти наборы в дальнейшем не смешиваются и для каждой выходной переменной строится своя сеть. Если мы считаем выходы зависимыми (т.е. подчиняющимися какой-то общей закономерности), то на каждом ряду селекции переменные смешиваются, т.е. в отбор включаются линейные комбинации переменных, отобранные и по разным выходам.

Кроме того, в данном параграфе приведён двойной генетический алгоритм построения коллектива нейронных сетей, суть которого состоит в генетическом алгоритме разбиения выборки на кластеры и генетическом алгоритме построения сети для каждого кластера.

Во втором параграфе данной главы строятся алгоритмы определения структуры сетей Кохонена и Гроссберга. В частности, приведён генетический алгоритм, при этом основные операции определяются естественным для данных сетей образом, и приводится многорядный алгоритм определения структуры сети Кохонена с различными вариантами реализации генетических операций.

В третьем параграфе второй главы анализируются особенности обучения RBF-сетей, стандартное представление которых имеет вид (7), причём  $\varphi_j(\|\mathbf{x} - \mathbf{c}_j\|) = \varphi(\alpha_j, \|\mathbf{x} - \mathbf{c}_j\|)$ .

Главная особенность такой базисной функции состоит в том, что для неё все направления от центра  $\mathbf{c}_j$  одинаковы. Если моделируемые данные обладают таким свойством, то это - достоинство, если нет – то недостаток. В случае, когда не все направления равноправны, более привлекательными оказываются другие базисные функции, несколько вариантов которых рассматривается далее в данном параграфе. В одном из таких вариантов, который является принципиально новым, учитывается поведение функции вдоль лучей, выходящих из центра  $\mathbf{c}_i$ . В двумерном случае имеем  $\varphi_i = \varphi_i(\rho, \psi)$ , где  $(\rho, \psi)$  - полярные координаты вектора  $\mathbf{x} - \mathbf{c}_i$ ,  $\mathbf{x}(r, \varphi)$  - текущая точка. Типичные для метода конечных элементов функции получаются как частный случай такой RBF-сети при соответствующем выборе функции  $\varphi_i$ . Если положение границы элемента относительно его центра характеризуется некоторой функцией  $\rho = a_i(\psi)$ , тогда можно взять  $\varphi_i = (1 - \rho / a_i(\psi))_+$ , где, как

обычно, обозначено  $w_+ = \begin{cases} w, w \geq 0; \\ 0, w < 0 \end{cases}$ . В многомерном случае можно взять  $\varphi_i = \varphi_i(\rho, \mathbf{v})$ , где

$\rho = \|\mathbf{x} - \mathbf{c}_i\|$ , а  $\mathbf{v}$  - вектор на единичной сфере.

Кроме того, приводятся методы и алгоритмы построения таких сетей - метод обучения, основанный на кластеризации и главных компонентах, алгоритм кластеризации ошибок и алгоритм построения коллектива RBF-сетей.

Если многослойный персептрон или RBF – сеть по отдельности не позволяют построить адекватной модели, то можно применить гибридную сеть, сочетающую в себе и тот и другой вид сетей. Построить такую сеть можно разными способами. Самый простой – задать выход комбинированной сети суммой выходов сетей одного и другого вида. Наиболее подходящими задачами для подобной архитектуры являются такие, в которых логика, моделируемая многослойным персептроном, отражает основные особенности моделируемых данных, но точность получается недостаточной. Для таких задач вначале обучаем многослойный персептрон, при этом число нейронов у него не должно быть слишком большим, далее вычисляем ошибки аппроксимации и к полученной остаточной выборке применяем алгоритм кластеризации ошибок.

Следующий очевидный подход – подать выход RBF–сети на вход многослойного персептрона или, наоборот, подать выход многослойного персептрона на вход RBF–сети – не представляется особенно удачным, хотя первый вариант может быть полезен для неявной нечёткой кластеризации, в особенности, если RBF–сеть строится с помощью алгоритма кластеризации ошибок или похожего на него.

Для последнего типа задач более перспективным видом сети является функция, получающаяся при замене элементов матриц  $W_i$  в формуле (1) на RBF–сети. При этом сначала строим набор RBF–сетей, причём к каждой сети относим достаточно представительную часть кластеров, центры которых по возможности удалены друг от друга. Выходы RBF–сетей подаём на входы нейронов с активационной функцией персептронного типа и для выходов этих нейронов повторяем предыдущий шаг и т.д. столько раз, сколько слоёв должен иметь многослойный персептрон. Получившуюся сеть можно доучить каким-либо методом нелинейной оптимизации.

Результат, аналогичный коллективу RBF–сетей, может дать сеть, конструкция которой является двойственной к предыдущей. У такой сети веса  $w_j$ ,  $\alpha_j$  и  $c_j$  заменяются на выходы персептронов. После начальной генерации, для которой можно использовать метод главных компонент, сеть можно доучить.

Далее в этой главе мы рассматриваем несколько видов нейронных сетей, специально приспособленных для обработки временных рядов. Основная задача, решаемая этими сетями, – подбор по выборке  $\{x(n), y(n)\}, n = 1, \dots, N$  зависимости вида  $y(n) = f(\mathbf{w}, x(n), x(n-1), \dots, x(1))$ , где  $x(n)$  – входной, а  $y(n)$  – выходной вектор. При этом выборка может пополняться, т.е.  $N$  может расти. В частности, для построения такой зависимости рассматривается генетический алгоритм

определения структуры сети прямого распространения с временными задержками и частичной структурой связей и аналогичный многорядный алгоритм.

Другой способ уменьшить количество подбираемых параметров во временных сетях – не изменять каждый вес сам по себе, а задаться некоторой зависимостью между ними. В формуле (10) свёртку можно трактовать как некоторый фильтр. На эту идею опирается блочный генетический алгоритм определения структуры сети прямого распространения с временными задержками и частичной структурой связей.

В пятом параграфе рассматриваются подходы к решению поставленных в первой главе задач кластеризации, в которых классифицируемая выборка меняется со временем. Динамическая кластеризация использована в генетическом алгоритме построения коллектива нейронных сетей прямого распространения с временными задержками и двойном генетическом алгоритме построения коллектива таких нейронных сетей.

В шестом параграфе второй главы рассматриваются RBF-сети с временными задержками. Первая форма таких сетей позволяет моделировать ситуации, когда происходят некоторые события, действия которых ослабевают со временем. Вторая форма временных RBF-сетей предназначена для моделирования динамики систем с несколькими взаимодействующими центрами. Для одной и другой формы сетей приведён ряд методов и алгоритмов, позволяющих подобрать как их веса, так и параметры.

В седьмом параграфе изучены рекуррентные сети, приведены возможные постановки задачи обучения с комментариями соответствующих им особенностей и построены методы и алгоритмы определения структуры решающих конкретный вариант задачи нейронных сетей и их весов. Варианты постановок задач:

1) Сеть – и структура и веса – фиксируется заранее, само обучение не происходит, результат работы сети – просто результат её функционирования.

2) Есть ряд временных последовательностей – как входных, так и соответствующих им выходных, надо обучить сеть строить по входной последовательности выходную.

3) Задан ряд временных последовательностей, с которыми сеть работает в режиме самообучения, например, образует кластеры.

4) Есть некоторый объект, моделью которого должна стать сеть. Мы можем подавать на вход объекта некоторую временную последовательность воздействий и получать соответствующую ей выходную. При этом процедура построения обучающей выборки зависит от целей, которые мы ставим:

- Одной из таких целей может быть моделирование объекта при экстремальных значениях выхода, при этом входная последовательность подбирается так, чтобы достичь этих экстремальных значений.

- Другой целью может быть моделирование объекта при выходах, находящихся в определённой области. При этом в обучающее множество включаются те наборы из элементарных последовательностей, которые соответствуют требуемым выходам.

- Третьей целью может быть максимально равномерное уменьшение ошибки моделирования. Для её достижения можно пополнять обучающую выборку векторами из тех областей входного пространства, которые соответствуют максимальным ошибкам (разности между выходом сети и выходом моделируемого объекта). Для этого можно применить метод кластеризации ошибок в рассматриваемом подмножестве пространства входных векторов, т.е. множество пар вида  $(\mathbf{x}(n), y(n) - \hat{y}(n))$  кластеризуется, и центры кластеров, соответствующие максимальным ошибкам, берутся в качестве новых входных векторов.

5) Поступает одна входная временная последовательность векторов и одна выходная. Задача состоит в том, чтобы в процессе поступления данных настроить (а при необходимости перестроить) веса сети таким образом, чтобы сеть наилучшим образом отражала зависимость между входными и выходными данными.

6) Если сеть моделирует некоторый объект во время его функционирования, то мы имеем ту же самую задачу, только возникает вопрос об оптимальном формировании входных сигналов. Одним из подходов к такому формированию может быть работа на уровне отрезков входной последовательности, длительность которых сравнима со временем реакции объекта. При этом отрезки, для которых есть основание ожидать наибольших ошибок, включаются в выборку с большей вероятностью.

7) Поступает одна последовательность векторов, которая обрабатывается сетью в процессе самообучения. Целью такой обработки может быть, к примеру, динамическая кластеризация, при которой сеть, в отличие от пункта 3, разбивает на классы не последовательности, а возникающие ситуации в одной поступающей последовательности.

8) Отдельного анализа требует работа с управляемым объектом. Если модель управляемого объекта заранее неизвестна, тогда можно идти двумя путями. Во-первых, можно непосредственно подбирать управление объектом в процессе его функционирования в том или ином классе нейросетевых функций, адаптируя это управление в соответствии с полученными результатами. Во-вторых, можно по собранным данным строить модель или набор моделей и подбирать управление, оптимальное для текущей лучшей модели и удовлетворяющее необходимым требованиям для всей совокупности рассматриваемых моделей (или некоторой её части). В такой постановке параметры и структура моделей объекта подбирается по экспериментальным данным, а структура и параметры управления – исходя из оптимизации функционала, задаваемого целями управления.

В восьмом параграфе второй главы решается вопрос адаптации структуры сети в случае выборки, пополняемой в разном темпе. Дело в том, что генетические алгоритмы остаются

достаточно медленной процедурой, на которую может не хватить времени. Особенно актуальной становится эта проблема, когда периодически приходят новые наблюдения, которые нужно включать в обработку. При этом существенное значение имеет не только временной интервал прихода этих наблюдений  $\delta T$ , но и характерное время перестройки изучаемого процесса  $\Delta T$ . Анализируются разные ситуации и модификации построенных ранее алгоритмов для них.

С помощью методов второй главы можно решать разнообразные достаточно сложные задачи математического моделирования. Один из классов таких задач, связанных с дифференциальными уравнениями, в первую очередь с задачами математической физики, рассматривается в **третьей главе**. Кроме того, в третьей главе поставлена задача построения робастной математической модели по разнородным данным, включающим как уравнения, так и экспериментальные наблюдения. Указаны новые примеры подобных задач для дифференциальных и некоторых других уравнений и предложена общая методология их решения в рамках нейросетевой парадигмы. На основе разработанных в диссертации общих принципов созданы методы решения классических и некоторых неклассических задач математической физики:

- Метод кластеризации ошибок в применении к решению краевой задачи для уравнения Лапласа и реализующий его алгоритм.
- Метод «Внутренние круги» и реализующий его алгоритм.
- Метод кластеризации ошибок в применении к решению краевой задачи общего вида и реализующий его алгоритм.
- Метод решения задачи Стефана с помощью рекуррентных нейронных сетей и реализующий его алгоритм.
- Метод решения задачи о датчике переменного давления с помощью системы нейронных сетей и реализующий его алгоритм.
- Генетический метод построения нейронной сети для решения задачи, допускающей декомпозицию, и реализующий его алгоритм.
- Процедура кластеризации ошибок для задачи, допускающей декомпозицию, и реализующий её алгоритм.
- Генетический метод построения ансамбля нейронных сетей для решения задачи, допускающей декомпозицию, и реализующий его алгоритм.
- Многорядный метод построения нейронной сети для решения задачи, допускающей декомпозицию, и реализующий его алгоритм.
- Обобщённый метод кластеризации ошибок и реализующий его алгоритм.
- Обобщённый многорядный метод и реализующий его алгоритм.

В первом параграфе третьей главы проанализированы классические и неклассические задачи для обыкновенных дифференциальных уравнений. Простейшим примером неклассической задачи

является практически не рассматривавшаяся ранее модификация задачи Коши для уравнения  $y'(x) = F(x, y)$ , состоящая в замене начального условия набором условий (обычно это результаты наблюдений)  $y(\tilde{x}_1) = f_1, \dots, y(\tilde{x}_p) = f_p$ . Её приближённое решение на промежутке  $(a; b)$  предложено

искать минимизацией функционала  $J(y) = \int_a^b |y'(x) - F(x, y)|^2 dx + \delta \sum_{k=1}^p |y(\tilde{x}_k) - f_k|^2 \rightarrow \min$  на

некотором множестве функций, например, нейросетевых. На практике обычно используется дискретное представление функционала ошибки вида

$$J(y) = \sum_{j=1}^M |y'(x_j) - F(x_j, y(x_j))|^2 + \delta \sum_{k=1}^p |y(\tilde{x}_k) - f_k|^2, \quad (17)$$

где  $\{(x_j)\}_{j=1}^M$  - множество тестовых точек на интервале  $(a; b)$ , которое меняется в процессе обучения.

При этом погрешности в данных мало влияют на конструируемое нейросетевое решение. Заметим, что некоторые из точек  $\tilde{x}_k$  могут и не принадлежать промежутку  $(a; b)$ .

Обычно применяемый в западной литературе для поиска приближённого нейросетевого решения классической задачи Коши или краевой задачи подбор весов сети при фиксированном наборе контрольных точек часто приводит к переобучению, когда малые ошибки в точках тестового множества сопровождаются большими ошибками вне него. Перегенерация тестовых точек делает процесс обучения сети более устойчивым. Если она производится на каждом шаге, тогда процесс обучения слишком сильно замедляется. Чересчур редкая регенерация приводит к тому, что сеть успевает переобучиться, и на новом множестве точек обучение начинается практически заново. Разумный выбор частоты регенерации позволяет избежать и той, и другой опасности.

Если нужно подобрать не только веса нейронной сети, но и её топологию, т.е. структуру модели, то можно применить один из структурных алгоритмов, рассмотренных во второй главе. Если решение ищется в виде RBF-сети, то проще всего применить алгоритм кластеризации ошибок. Его особенность для рассматриваемой задачи состоит в том, что на каждом шаге алгоритма возникают новые тестовые точки, которые можно брать гуще там, где есть основание ожидать больших ошибок, например, в окрестности точек, в которых ошибки предыдущего шага максимальны. Решая задачу с помощью нейронной сети другого вида, можно применить какой-либо из генетических алгоритмов. Работу такого алгоритма облегчает возможность выбора новых тестовых точек и вычисление соответствующего функционала, который можно использовать для выбора наилучших из обученных сетей. Применение персептронов обычно оказывается оправданным, если в задаче присутствуют резкие переходы или есть основания ожидать, что такие резкие переходы присутствуют в решении.

С точки зрения нейросетевого подхода ненамного более сложной является задача поиска функции  $F(x, y)$  по результатам наблюдений в виде нейросетевого разложения

$$F_N(x, y) = \sum_{i=1}^N h_i v(x, y; \mathbf{a}_i), \text{ при этом одновременно строится уравнение и его решение.}$$

Нейросетевая методология позволяет изучить естественные обобщения рассмотренных выше подходов на системы обыкновенных дифференциальных уравнений и уравнения более высокого порядка. Например, приближённое решение краевой задачи для стационарного оператора Шредингера можно искать в виде нейронной сети, обучающейся на основе минимизации функционала ошибки

$$J(y) = \sum_{j=1}^M |y''(x_j) + q(x_j)y(x_j)|^2 + \delta(|y(x_0) - y_0|^2 + |y(x_1) - y_1|^2). \text{ Интересной для}$$

приложений является и задача определения потенциала  $q(x)$  по данным наблюдений, которая решается так же, как и приведённая выше задача поиска  $F(x, y)$ .

Аналогично решается и задача построения по экспериментальным данным уравнения второго порядка  $y''(x) = F(x, y, y')$ . В этом случае также можно использовать представленный выше подход, выбрав соответствующие нейросетевые функции. Следует особо отметить тот факт, что существенных изменений в представленные выше алгоритмы вносить не требуется.

Принципиально новый класс моделей возникает, если применить многорядный алгоритм, у которого новый ряд селекции определяется равенством  $y_{l+1} = w_i y_{i,l} + w_j y_{j,l} + w_{\varphi,k} \varphi(y_{k,l}) + w_{d,k} \frac{d}{dx}(y_{k,l})$ , где  $\varphi$  - некоторая нелинейная активационная функция. Более сложные модели возникают, если  $w_i, w_j, w_{\varphi,k}$  и  $w_{d,k}$  считать не константами, а функциями, которые можно представить нейронными сетями. Для определения структуры таких моделей предложен дважды многорядный алгоритм. Очевидно, что можно сразу рассмотреть многорядную модель, когда на вход слоя нейронов подаётся результат действия линейного дифференциального оператора на выходы предыдущего слоя, а функция активации может быть просто сигмоидной функцией или некоторым нелинейным оператором (линейным в малом и имеющим ограниченный образ в большом).

В следующих параграфах данной главы приведённые методы распространяются с обыкновенных дифференциальных уравнений на дифференциальные уравнения с частными производными.

Во втором параграфе в качестве модельной рассматривается задача Дирихле для уравнения Лапласа в единичном круге  $\Delta u = 0$  при  $r < 1$ ,  $u = f$  при  $r = 1$ . Ищем её решение в виде

$$u(x, y) = \sum_{i=1}^N c_i \exp\left\{-a_i \left[(x - x_i)^2 + (y - y_i)^2\right]\right\}. \text{ В качестве минимизируемого функционала выбираем}$$

$$J(u) = \int_{\Omega} |\Delta u|^2 d\Omega + \delta \int_{\Gamma} |u - f|^2 d\Gamma \text{ или его дискретный аналог.}$$

Важное преимущество нейросетевого подхода – достаточно развитая методика подбора оптимальной структуры сети. К решению рассматриваемой задачи предлагается применить следующий вариант метода кластеризации ошибок:

1. Выбирается достаточно небольшое число базисных функций, и строится начальное приближение путём подбора весов сети.

2. Вычисляются ошибки  $z_i$  на некотором множестве точек  $\{x_i, y_i\}_{i=1}^m$  границы и лапласиан  $z_i = \Delta u(x_i, y_i)$  в случайном наборе точек  $\{x_i, y_i\}_{i=m+1}^{m+M}$  внутри круга.

3. Происходит кластеризация точек  $\{x_i, y_i, z_i\}_{i=1}^{m+M}$  в соответствующем трёхмерном пространстве.

4. Выбираются кластеры, соответствующие максимальным средним по кластеру ошибкам, и строится приближение к решению для данного кластера с помощью соответствующих нейросетевых функций.

5. Коэффициенты общего набора уточняются исходя из условия минимизации соответствующего функционала.

6. Процедура повторяется необходимое число раз.

При решении этой задачи, в частности выяснилось, что подбор входящих нелинейно параметров позволяет в три раза уменьшить общее число параметров, необходимое для достижения одной и той же точности по сравнению с подбором только параметров  $c_i$ . Это означает сокращение числа требуемых функций на порядок.

Далее в этом параграфе разработано распространение нейросетевого подхода на решение аналогичных задач в случае более высоких размерностей и областей более общего вида. Наряду с использованием RBF-сетей в некоторых задачах более эффективно использование нейронных сетей другого типа. Можно для поиска решения применить персептрон с несколькими скрытыми слоями или какую-либо гибридную сеть. Аналогично рассматриваются и обобщения: линейное

$u = \zeta_0 + \sum_{i=1}^p \zeta_i \cdot x_i$  или квадратичное  $u = \zeta_0 + \sum_{i=1}^p \zeta_i \cdot x_i + \sum_{i=1}^p \sum_{j=1}^p \zeta_{ij} \cdot x_i x_j$ , где коэффициенты  $\zeta$  -

персептроны.

Интересное обобщение, если в качестве входов сети использовать не только  $x$ ,  $y$  и  $z$ , но и граничные значения  $u$  в некотором наборе точек. Обученная таким образом нейронная сеть позволяет получить решение задачи Дирихле не при фиксированных, а при произвольных граничных условиях (функция  $f$  задаётся таблично в этом наборе точек). Аналогично можно поставить и решить обратные задачи разного рода – например, определить граничные условия по решению, задаваемому в некотором наборе точек.

Выше мы почти не затрагивали задачи, зависящие от времени. Казалось бы, что для таких задач логично использовать рекуррентные сети, однако это соответствие не так однозначно. С одной



стороны, время можно рассматривать в качестве такой же координаты, как и  $x$ ,  $y$  и  $z$ , применяя предложенные выше подходы. С другой стороны, можно обучать рекуррентную сеть моделировать переход от одного слоя к другому не только по времени, но и по пространству (например, от поверхности шара к внутренним шаровым слоям). Возможен и такой подход, при котором решение задачи ищется в виде нейронной сети с параметрами, зависящими от времени.

В третьем параграфе разработаны методы решения характерных для описания многокомпонентных систем краевых задач следующего вида: в области  $\Omega = \bigcup_i \Omega_i$  найти функцию  $u$ , заданную кусочно:  $u = u_i(\mathbf{x})$  при  $\mathbf{x} \in \Omega_i \subset R^p$ , так, что её сужение  $u_i$  удовлетворяет в подобласти  $\Omega_i$  уравнению  $L_i(u_i) = g_i$ , где  $L_i$ - дифференциальный оператор с частными производными, и краевым условиям  $B_j(u_j)|_{\Gamma_j} = f_j$ ,  $\partial\Omega = \Gamma = \bigcup_j \Gamma_j$ , различные компоненты решения - условиям согласования  $C_{kl}(u_k)|_{\Gamma_{kl}} = C_{kl}(u_l)|_{\Gamma_{kl}}$ , где  $\Gamma_{kl}$  участок стыка подобластей  $\Omega_k$  и  $\Omega_l$ .

Для решения поставленной задачи предложено два принципиально различных подхода. При первом - ищется сеть, дающая приближённое решение во всей области. Достоинством такого подхода является простота реализации и бесконечная гладкость полученного решения в случае выбора соответствующих функций активации. Главный недостаток состоит в том, что мы пытаемся точные решения, которые могут быть разрывными или у которых разрывными являются первые или вторые производные, приблизить бесконечно гладкими функциями – в такой ситуации не следует ожидать очень хорошей точности. При втором подходе для каждой подобласти строится своя сеть. Достоинством такого подхода является большая точность аппроксимации для каждой подобласти при фиксированном числе нейронов, недостатком – необходимость стыковать решения между собой, что влечёт усложнение алгоритма. При этом стыковку можно производить, либо добавляя соответствующее слагаемое в функционал и обучая всю совокупность сетей сразу, либо чередуя процессы раздельного обучения сетей с процедурой их стыковки, построенной, например, на основе альтернирующего метода Шварца.

В качестве примера нестандартной постановки решена задача нахождения функции, для которой в некоторой части области известно уравнение, кроме того, известны (например, в результате измерений) её значения в некотором наборе точек.

Кроме того, разработаны конкретные приложения изложенных ранее идей и методов к задаче расчёта теплообмена в системе «сосуды-ткани», решение которой представляет определённые вычислительные сложности в связи с присутствием в ней нескольких компонент и разным масштабом их измерений.

Рассмотрение многокомпонентных систем существенно усложняется в случае, когда граница раздела компонент должна быть найдена в процессе решения. Особенно сложной становится задача в

случае, когда присутствует фазовый переход, т.е. одна компонента переходит в другую. В качестве модельной в четвертом параграфе рассмотрена одномерная задача Стефана.

Предложены следующие естественные с точки зрения методологии нейронных сетей подходы к задаче Стефана:

- Первый подход - аппроксимация температурных полей для обеих фаз с помощью соответствующим образом обученной RBF-сети или персептрона.
- Второй подход состоит в построении гетерогенной сети, которая включает в себя наряду с RBF сетями, описывающими температурные режимы для каждой из фаз, еще и персептрон с одним скрытым слоем, задающий границу раздела. В этом варианте требуется меньшее суммарное количество функций, хотя вычислительный процесс оказывается менее устойчивым.
- Третий подход состоит в поиске температурного поля с помощью пространственной RBF-сети (т.е. сети, входом которой является переменная  $x$ ), зависящие от времени веса которой находятся из системы обыкновенных дифференциальных уравнений.
- Четвертый подход даёт использование рекуррентных нейронных сетей для задания сеточной аппроксимации нестационарных температурных режимов фаз.

Далее решается задача моделирования поверочной камеры калибратора переменного давления с помощью системы нейронных сетей. Особенностью этой задачи является то, что решение в области ищется в виде одной нейронной сети в процессе оптимизации одного функционала, а граница области – в виде другой нейронной сети и в процессе оптимизации другого функционала.

В пятом параграфе рассматривается аппроксимация решения задачи Дирихле для уравнения Лапласа в области специального вида, допускающей декомпозицию, с помощью нейронных сетей, для обучения которых приводятся оригинальные алгоритмы, основанные на идеях эволюционного моделирования. Приведенные алгоритмы допускают эффективное распараллеливание и легко обобщаются на другие подобные задачи. Численные эксперименты показали, что применение алгоритмов эволюционного типа позволяет сократить требуемое число функций (нейронов) в 3-10 раз при сохранении точности, при этом скорость обучения увеличивается в 2-4 раза.

Так же, как и для обыкновенных уравнений, можно строить модель в виде уравнения в частных производных по данным измерений, определяя коэффициенты этой модели как некоторые нейросетевые функции. Другое обобщение – на случай систем уравнений (не только дифференциальных) – также не вызывает особых трудностей. В шестом параграфе описана достаточно общая постановка такой задачи и предложены подходы к её решению. Изучение модельной задачи показало, что нейросетевой подход позволяет решать задачи такого типа достаточно эффективно.

Аналогично тому, как это было сделано для обыкновенных дифференциальных уравнений, можно рассмотреть и многослойную модель с частными производными. В частности, многорядный

алгоритм построения такой модели практически не отличается от рассмотренного выше, только на каждом ряду селекции следует рассматривать модели вида

$$y_{l+1} = w_i y_{i,l} + w_j y_{j,l} + w_{\varphi,k} \varphi(y_{k,l}) + w_{p,k} \frac{\partial}{\partial x_p} (y_{k,l}).$$

Можно так же, как и выше, рассмотреть временную модель с частными производными и задачу управления объектом, который можно описать такой моделью, однако принципиальных отличий по сравнению с приведёнными выше рассуждениями решение задач такого рода не имеет.

Приведённые в данной главе многочисленные результаты конкретных расчётов позволяют утверждать, что нейронные сети являются эффективным средством решения задач такого рода.

Как было показано ранее, обучение нейронных сетей обычно сводится к некоторой итерационной процедуре оптимизации нелинейного функционала. Поэтому условия сходимости такого рода процедур представляют большой интерес. Особенно интересными являются проанализированные в **четвёртой главе** итерационные процессы, обобщающие метод Ньютона, в силу своей быстрой сходимости. Стандартный метод Ньютона не совсем удобен в силу двух причин – локальной сходимости и больших вычислительных затрат при решении линеаризованной системы на каждом шаге. Эти причины побуждают рассматривать более сложные подходы, которые в значительном числе задач оказываются более эффективными. Главной особенностью доказанных в диссертации теорем о сходимости итерационных процессов по сравнению с известными результатами является рассмотрение процессов в последовательности пространств, вложенных друг в друга, сочетание шагов методов разных порядков, рассмотрение оценки Гёльдера с переменным показателем и замена обратного оператора приближённым обратным с оценкой, меняющейся от шага к шагу.

Метод Ньютона является локальным, т.е., применяя его к задаче нахождения экстремума функционала ошибки, можно получить только локальный экстремум, который может и не быть глобальным. Для преодоления этих трудностей во втором параграфе четвёртой главы дается краткое рассмотрение нескольких модификаций рассмотренных в диссертации алгоритмов обучения, позволяющих приблизиться к глобальному экстремуму. Эти подходы были проверены в задачах обучения нейронных сетей и оказались достаточно эффективными в случае, когда число подбираемых переменных (весов сети) составляет от сотен до нескольких тысяч.

В последних трёх параграфах четвёртой главы разработаны подходы к распределённой реализации рассмотренных в диссертации алгоритмов. В качестве основной области их применения может быть указана типичная для Интернета задача обработки информации в ситуации, когда связи между узлами ненадёжны, а пересылка всех данных в один узел невозможна или нерациональна. Работа с нейронными сетями в такой ситуации возможна в двух вариантах, проанализированных в

данной главе: либо единая сеть обучается на многих компьютерах параллельно, либо обучается и работает сеть, отдельные части которой находятся на разных компьютерах.

В связи с этим, в третьем параграфе разработаны методы и алгоритмы распределённого обучения нейронных сетей, в четвёртом - методы и алгоритмы обучения нейронных сетей по распределённым данным и задача построения и обучения распределённых нейронных сетей, т.е. таких сетей, у которых отдельные части находятся на различных компьютерах.

**В пятой главе** изучаются осцилляторные нейросетевые модели бесконечной размерности. При этом место квазипериодических колебаний, рассматриваемых в известных нейросетевых работах, занимают почти периодические колебания с бесконечным базисом частот. В данной главе введены некоторые математические конструкции, которые могут служить основой для разработки теории таких колебаний.

Квазипериодическое движение конечномерной системы можно трактовать как движение по некоторому тору соответствующей размерности. Если система имеет бесконечную размерность, то можно ожидать почти периодического движения с бесконечным базисом частот, которое можно рассматривать как движение по тору бесконечной размерности. Один из результатов, который получен в первом параграфе, состоит в сохранении характера движения при малом возмущении.

Если рассмотреть окрестность тора из предыдущего параграфа, то при достаточной гладкости поведение исходной системы будет близко к поведению соответствующей линеаризации. Для изучения поведения системы важно привести линейную часть к некоторому каноническому виду. Наибольший интерес и сложность представляет собой резонансный случай, частным вариантом которого является система с нечётными почти периодическими коэффициентами. Эта задача решена во втором параграфе.

Прежде чем рассматривать систему нелинейных уравнений с почти периодическими коэффициентами, логично остановиться на случае одного уравнения. В третьем параграфе рассматривается поиск почти периодических решений аналитического уравнения с почти периодическими коэффициентами. Распространение этого результата на общий случай уравнения в банаховом пространстве принципиальных трудностей не вызывает.

Все эти результаты получаются применением метода последовательных замен, который сформулирован и изучен в общем виде в последних трёх параграфах данной главы. В последнем параграфе получены условия его сходимости, следствием которых и являются результаты первых трёх параграфов. Формулировка соответствующей теоремы потребовала обобщения понятия банаховой группы Ли на случай, когда координатное пространство является системой вложенных друг в друга банаховых пространств. При этом соответствующая теорема приобрела форму утверждения об относительной локальной транзитивности соответствующего действия.

Обоснованный в данной главе метод последовательных замен можно применить и как эффективную вычислительную процедуру исследования квазипериодических колебаний систем конечной размерности, что позволяет строить явные приближённые решения таких систем.

Нейросетевой эмулятор Essence, описанию которого посвящена **шестая глава**, разработан EssenceGroup под руководством автора. Текущая версия (Essence 1.2) представляет собой написанную на базе платформы Java 2 программную среду, которая предназначена для разработки и обучения нейронных сетей с целью отыскания скрытых зависимостей в числовых данных.

В законченной версии программы реализована наиболее употребительная архитектура нейронных сетей – многослойный персептрон. При этом на число слоев и число нейронов в каждом слое не накладывается никаких ограничений. Каждый слой может быть гибко настроен. Так, например, можно не только изменить количество нейронов в слое, но и указать типы нейронов. Предусмотрено девять видов функций активации нейронов, любое количество слоев и возможность устанавливать отдельно число и вид нейронов для каждого слоя и т. д.

Достоинством программы является:

- программа без перекомпиляции может быть запущена на любой платформе, на которой установлена виртуальная машина Java;
- использование технологии JavaBeans позволяет создавать и распространять обученную нейронную сеть в виде отдельного легковесного компонента или апплета;
- использование XML в качестве формата хранения и передачи данных позволяет упростить работу и взаимодействие Essence с другими программами;
- наличие развитого пользовательского интерфейса, привычного для пользователей и обладающего гибкостью и возможностью расширения за счет встраиваемых модулей.

Мощность и потенциал платформы Java, её расширенные возможности при работе с сетью, базами данных, а также поддержка наиболее передовых технологий и тенденции ее развития позволяют в перспективе перейти от локальных версий пакета Essence к сетевым, распределенным, что позволит максимально полно использовать преимущества нейронных сетей при работе в параллельном режиме и перейти к обработке распределённых по Интернет данных.

В третьем и четвёртом параграфе данной главы приведены результаты некоторых конкретных прикладных исследований, выполненных с помощью Essence. Эти исследования связаны с построением модели климатических характеристик Западной Сибири. Данная модель использовалась для описания условий работы техники в данном регионе, она позволила сделать важные выводы о необходимых конструктивных особенностях рассматриваемых машин, в частности, рассчитать для них необходимую мощность средств тепловой предпусковой подготовки.

### 3. ЗАКЛЮЧЕНИЕ

Итогом диссертационной работы являются следующие научные и практические результаты:

1) Большинство используемых в настоящее время архитектур нейронных сетей описаны с помощью единого формализма, что позволяет единообразно применять на этапах подбора весов нейронных сетей при реализации разрабатываемых алгоритмов известные методы нелинейной оптимизации.

2) Создана методология конструирования и обучения для широкого класса нейронных сетей, применимая к различным задачам математического моделирования.

3) На основе созданной методологии разработана серия методов и реализующих их алгоритмов, позволяющая единообразно для упомянутых в предыдущем пункте нейронных сетей подбирать оптимальную структуру в зависимости от решаемой задачи моделирования.

4) Представлен единый подход, позволяющий искать в нейросетевых функциональных пространствах приближённое решение обыкновенных дифференциальных уравнений, дифференциальных уравнений с частными производными и иных уравнений подобного рода.

5) Сформулированы алгоритмы построения с помощью нейронных сетей моделей в виде, как классических дифференциальных уравнений, так и уравнений нового типа – многослойных нейродифференциальных уравнений.

6) Доказана сходимость итерационных процессов, обобщающих метод Ньютона, что позволяет изучать условия сходимости рассмотренных в диссертации процедур обучения нейронных сетей. Главными особенностями доказанных в диссертации теорем по сравнению с известными результатами является рассмотрение итерационного процесса в последовательности пространств, вложенных друг в друга, сочетание шагов методов разных порядков, рассмотрение оценки Гёльдера с переменным показателем и замена обратного оператора приближённым обратным с оценкой, меняющейся от шага к шагу.

7) Предложены новые варианты алгоритмов глобальной оптимизации, позволяющих приближаться к глобально оптимальным весам нейронных сетей в процессе их обучения. Эти алгоритмы оказались эффективными при подборе весов нейронных сетей в пространствах размерностей от нескольких сотен до нескольких тысяч.

8) Приведены новые распределённые варианты алгоритмов обучения нейронных сетей, позволяющие проводить построение таких сетей в рамках парадигмы Интернет-вычислений.

9) Предложены подходы к изучению осцилляторных моделей бесконечной размерности и с их помощью доказаны теоремы об устойчивости почти периодических колебаний соответствующих систем, что позволяет подойти к изучению моделей мозга, включающих реальное число нейронов.

10) Разработан нейросетевой эмулятор Essence, позволяющий изучать алгоритмы обучения нейронных сетей и решать практические задачи применения нейросетевых моделей. Кроме того, на основе Essence может быть построен распределённый комплекс программ, работающий в среде Интернет.

### СПИСОК ПУБЛИКАЦИЙ ПО ТЕМЕ ДИССЕРТАЦИИ

1. Тархов Д.А. О почти периодическом возмущении на бесконечномерном торе // Изв. АН СССР сер. матем. 1986. т. 50 №3 с.617-632.
2. Тархов Д.А. Интегрирование почти периодических функций и бесконечномерная теория диофантовых приближений // Матем. заметки 1990 т. 47 №5 с.106-115.
3. Тархов Д.А. О приводимости линейного дифференциального уравнения с нечётными почти периодическими коэффициентами // Функ. Анализ и его прил. 1994 т.28 №1 с. 82-85.
4. Тархов Д.А. Приводимость линейной системы дифференциальных уравнений с нечётными почти периодическими коэффициентами // Матем. заметки 1996 т. 60 №1 с.109-119.
5. Тархов Д.А. Нейронные сети: модели и алгоритмы. М.: Радиотехника, 2005. 256с.
6. Тархов Д.А. Нейронные сети как средство математического моделирования М.: Радиотехника, 2006, Нейрокомпьютеры: разработка, применение, 2006, №2.
7. Васильев А.Н., Тархов Д.А., Новые подходы на основе RBF – сетей к решению краевых задач для уравнения Лапласа на плоскости // Нейрокомпьютеры: разработка, применение. - М.: Радиотехника, 2004, №7-8. с. 119-126.
8. Васильев А.Н., Тархов Д.А., Нейронные сети как новый универсальный подход к численному решению задач математической физики // Нейрокомпьютеры: разработка, применение. — М.: Радиотехника, 2004, №7-8. с.111-118.
9. Васильев А.Н., Тархов Д.А., Нейросетевые подходы к решению краевых задач в многомерных составных областях // Известия ТРТУ. - 2004, №9. с.80-89.
10. Васильев А.Н., Тархов Д.А., Применение искусственных нейронных сетей к моделированию многокомпонентных систем со свободной границей // Известия ТРТУ. - 2004, №9. с.89-100.
11. Васильев А.Н., Тархов Д.А., Построение нейросетевой модели по дифференциальным уравнениям и экспериментальным данным // Известия ТРТУ. - 2005, № 10 с. 98-107.
12. Антонов В.И., Васильев А.Н., Тархов Д.А. Приближённое решение задачи Стефана с помощью искусственных нейронных сетей // Материалы международной конференции «Искусственный интеллект – 2004» т.1. – Таганрог – Донецк 2004 с. 405-408.
13. Васильев А.Н. Попов А.Г. Тархов Д.А. Нейросетевые технологии анализа данных и возможность их использования в современных АСУ ТП // Второй научно-технический семинар «Современные системы контроля и управления электрических станций и подстанций (АСУ ТП) на базе микропроцессорной техники». 2001.
14. Васильев А.Н., Тархов Д.А., Нейросетевой подход к решению некоторых неклассических задач математической физики // сборник научных трудов VII Всероссийской научно-технической конференции «Нейроинформатика-2005», часть 2, Москва, МИФИ, 2005, с.52-60.
15. Васильев А.Н., Тархов Д.А., Применение нейронных сетей к неклассическим задачам математической физики // сборник докладов Международной конференции по мягким вычислениям и измерениям – SCM'2003, том 1, с.337-340.
16. Васильев А.Н., Тархов Д.А. Моделирование распределённых систем с помощью нейронных сетей // Труды 5-й международной научно-технической конференции «Компьютерное моделирование 2004» СПб. 2004 с.171-172.
17. Васильев А.Н., Тархов Д.А., Гущин Г. Моделирование калибратора переменного давления с помощью системы нейронных сетей, Сборник докладов Международной конференции по

- мягким вычислениям и измерениям – SCM'2004, Санкт-Петербург, СПбГЭТУ «ЛЭТИ», том 1, с.304-308.
18. Васильев А.Н., Тархов Д.А. RBF-сети и некоторые задачи математической физики // Сборник докладов Международной конференции по мягким вычислениям и измерениям – SCM'2004, Санкт-Петербург, СПбГЭТУ «ЛЭТИ», том 1, с.309-312.
  19. Васильев А.Н., Тархов Д.А. Нейросетевой подход к решению краевых задач в составных областях // Материалы международной конференции «Искусственный интеллект – 2004» т.1. – Таганрог – Донецк 2004 с. 475-478.
  20. Васильев А.Н., Тархов Д.А. Новые нейросетевые подходы к решению краевых задач в составных областях // Искусственный интеллект. — Донецк 2005, №1 с. 26-36.
  21. Васильев А.Н., Тархов Д.А. Применение искусственных нейронных сетей к задаче Стефана // Искусственный интеллект. — Донецк 2005 №1 с. 37-47.
  22. Васильев А.Н., Тархов Д.А. Нейросетевая методология построения приближённых решений дифференциальных уравнений по экспериментальным данным // Материалы международной конференции «Интеллектуальные и многопроцессорные системы – 2005» т.2. – Таганрог – Донецк – Минск 2005 с. 219-223.
  23. Васильев А.Н., Тархов Д.А., Антонов В.И. Нейросетевой подход к моделированию теплообмена в системе сосуда-ткани // Материалы международной конференции «Интеллектуальные и многопроцессорные системы – 2005» т.2. – Таганрог – Донецк – Минск 2005 с. 223-227.
  24. Власов Л.В., Малыхина Г.Ф. Тархов Д.А. Нейросетевой эмулятор «Essence» // Датчики и системы. Сборник докладов международной конференции, том 3. 2002.
  25. Тархов Д.А. Выпрямление интегральных кривых системы дифференциальных уравнений на бесконечномерном торе // Краевые задачи математической физики и смежные вопросы теории функций 16, (зап. научн. семин. ЛОМИ, т. 136). Л., Наука, 1984, с. 181-183.
  26. Тархов Д.А. Некоторые подходы к построению распределённых нейросетевых моделей // Труды 5-й международной научно-технической конференции «Компьютерное моделирование 2004» СПб. 2004 с.170 -171.
  27. Тархов Д.А. Распределённые вычисления и нейронные сети // Сборник докладов Международной конференции по мягким вычислениям и измерениям – SCM'2004, Санкт-Петербург, СПбГЭТУ «ЛЭТИ», том 1, с.295-298.
  28. Vasilyev A., Tarkhov D., Guschin G., “Neural Networks Method in Pressure Gauge Modeling”, Proceedings of the 10<sup>th</sup> IMEKO TC7 International Symposium on Advances of Measurement Science, 2004, Saint-Petersburg, Russia, Vol.2, pp.275-279.
  29. Vasilyev A., Tarkhov D., Human motion simulation // Труды 5-й международной научно-технической конференции «Компьютерное моделирование 2004» СПб. 2004 с.174-175.
  30. Dmitry A. Tarkhov and Alexander N. Vasilyev, New Neural Network Technique to the Numerical Solution of Mathematical Physics Problems. I: Simple Problems, Optical Memory and Neural Networks (Information Optics), Allerton Press, Inc., Vol. 14, No. 1, 2005, pp. 59-72.
  31. Dmitry A. Tarkhov and Alexander N. Vasilyev, New Neural Network Technique to the Numerical Solution of Mathematical Physics Problems. II: Complicated and Nonstandard Problems, Optical Memory and Neural Networks (Information Optics), Allerton Press, Inc., Vol. 14, No. 2, 2005, pp. 97-122.