МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ УКАРИНЫ
ДОНЕЦКИЙ НАЦИОНАЛЬНЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
Кафедра ПМиИ

ОТЧЕТ О НИРС

Анализ применимости искусственных нейронных сетей для задач управления динамическими объектами

Руководитель
доц., канд. техн. наук ____________________________ О.И. Федяев
(подпись) (дата)

Разработал
студент гр. ПО-99 а ____________________________ А.Е. Тихонов
(подпись) (дата)

2004

РЕФЕРАТ

Пояснительная записка работы 33 с., 9 рис., 6 источников.

Объект исследования – искусственные нейронные сети.

Цель работы – сравнение эффективности управления классическими методами (ПИД – регулятор) и нейросетевым контроллером (искусственная нейронная сеть обратного распространения ошибки).

Результаты работы – показано превосходство нейросетевого контроллера по быстродействию.

НЕЙРОН, НЕЙРОСЕТЬ, УПРАВЛЕНИЕ, ПИД

СОДЕРЖАНИЕ

ВВЕДЕНИЕ

В последние годы среди специалистов в области теории автоматического управления наметился подъем интереса к новым нетрадиционным подходам, которые формируются вне рамок классической парадигмы, базирующейся на аппарате интегро-дифференциального исчисления.

В 1992 г. Япония провозгласила программу Real World Computing (вычисления в реальном мире). Речь идет прежде всего о том, чтобы дать вычислительным и управляющим системам возможность самостоятельно, без помощи переводчика-человека воспринимать воздействия внешнего мира и действовать в нем. Авторы программы огромную роль – до 40% ее содержания отводят исследованию естественных и созданию искусственных нейронных систем (ИНС).

Традиционной областью применения ИНС считаются задачи распознавания изображений, речи, радарных сигналов. И все же основной областью применения, скорее всего, станет использование ИНС в системах управления и поведения роботами.

ИНС сегодня – это не столько совокупность заимствованных из нейрофизиологии моделей параллельных вычислительных структур, сколько арена борьбы о природе интеллекта. Если первый этап становления ИНС можно охарактеризовать как попытку синтезировать из набора сравнительно просто функционирующих нейронов некоторую упорядоченную структуру, способную выполнять сложные нелинейные преобразования, то сейчас передний фронт исследований перемещается в область психологии. когнитивных наук. Задача состоит в том, чтобы понять, какие сверхструктуры нейронов и как могут моделировать элементарные акты мыслительной деятельности, а затем воплотить эти принципы в работу систем управления.

1 ИСКУССТВЕННЫЕ НЕЙРОННЫЕ СЕТИ

1.1 Биологический прототип

Центральная нервная система имеет клеточное строение. Единица – нервная клетка, нейрон. Сеть нейронов, образующая человеческий мозг, представляет собой высокоэффективную, комплексную, существенно параллельную систему обработки информации. Нервная система человека имеет ошеломляющую сложность. Около 10^{11} нейронов участвуют в примерно 10^{15} передающих связях. Несомненно, что мозг работает более эффективно чем любая вычислительная машина, созданная человеком. Именно этот факт в течении многих лет побуждает и направляет работы ученых по созданию и исследованию искусственных нейронных сетей.

Нейрон имеет следующие основные свойства:

1. Участвует в обмене веществ и рассеивает энергию. Меняет внутреннее состояние с течением времени, реагирует на входные сигналы и формирует выходные воздействия и поэтому является активной динамической системой.
2. Имеет множество синапсов – контактов для передачи информации.
3. Нейрон взаимодействует путем обмена электрохимическими сигналами двух видов: электротоническими (с затуханием) и нервными импульсами (спайками), распространяющимися без затухания.

Существует два подхода к созданию искусственных нейронных сетей. Информационный подход: безразлично, какие механизмы лежат в основе работы искусственных нейронных сетей, важно лишь, чтобы при решении задач информационные процессы в НС были подобны биологическим. Биологический подход: при моделировании важно полное биоподобие, и необходимо детально изучить работу биологического нейрона.

На рис. 1.1 показана структура пары типичных биологических нейронов. Дендриты идут от тела нервной клетки к другим нейронам, где они принимают сигналы в точках соединения, называемых синапсами. Принятые синапсом входные сигналы подводятся к телу нейрона. Здесь они суммируются, причем одни входы стремятся возбудить нейрон, другие – воспрепятствовать его возбуждению. Когда суммарное возбуждение в теле нейрона превышает некоторый порог, нейрон возбуждается, посылая по аксону сигнал другим нейронам.

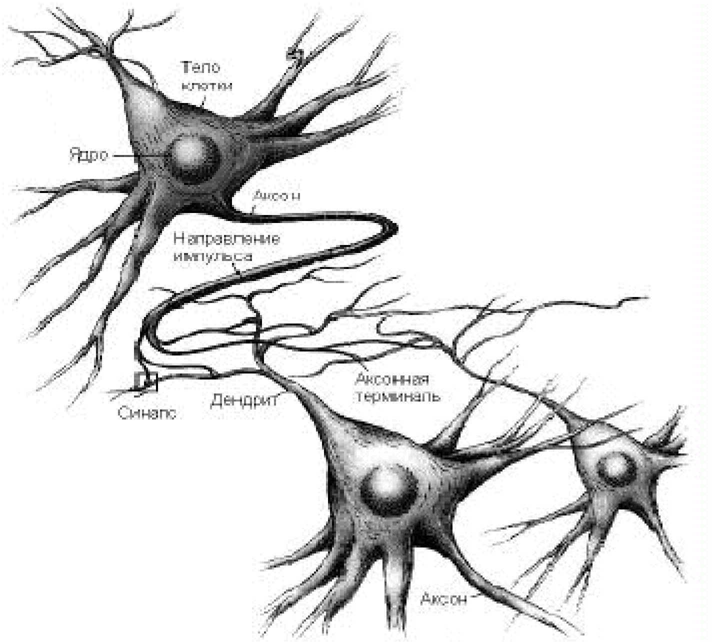

Рисунок 1.1 Естественная нейронная сеть

У этой основной функциональной схемы много усложнений и исключений, тем не менее большинство искусственных нейронных сетей моделируют лишь эти простые свойства.

1.2 Архитектура искусственных нейронных сетей

Рассмотрим элементы искусственных нейронных сетей:

1. Адаптивный сумматор вычисляет скалярное произведение вектора входного сигнала x на вектор параметров α. Адаптивным он называется из-за наличия вектора настраиваемых параметров α (рис.2.2). Для многих

задач полезно иметь линейную неоднородную функцию выходных сигналов. Ее вычисление также можно представить с помощью адаптивного сумматора, имеющего *n+1* вход и получающего на *0*-й вход постоянный единичный сигнал (рис.1.3).

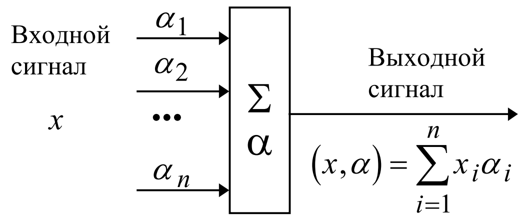

Рисунок 1.2 Адаптивный сумматор

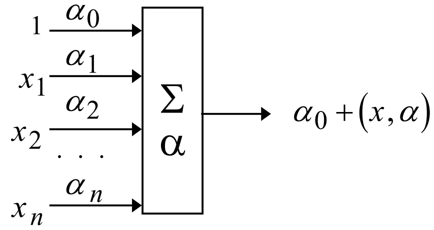

Рисунок 1.3 Неоднородный адаптивный сумматор

2. Нелинейный преобразователь сигнала – получает скалярный входной сигнал x и переводит его в $\varphi(x)$(рис.1.4).

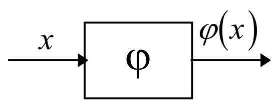

Рисунок 1.4 Нелинейный преобразователь сигнала

3. Точка ветвления служит для рассылки одного сигнала по нескольким адресам. Она получает скалярный входной сигнал x и передает его всем своим выходам (рис.1.5).

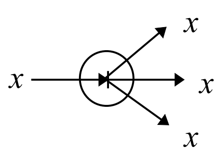

Рисунок 1.5 Точка ветвления

Соединяя адаптивный сумматор, нелинейный преобразователь и точку ветвления получаем формальный нейрон (рис.1.6).

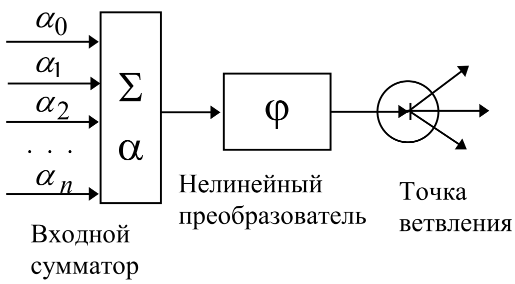

Рисунок 1.6 Формальный нейрон

Из формальных нейронов можно составлять слои нейронов, которые в свою очередь можно объединять в многослойные сети (рис.1.7). Нейроны первого слоя получают входные сигналы, преобразуют их и через точки ветвления передают нейронам второго слоя. Далее срабатывает второй слой и т.д. до *k*-го слоя, который выдает выходные сигналы для интерпретатора и пользователя. Если не оговорено противное, то каждый выходной сигнал *i*-го слоя подается на вход всех нейронов *i*+1-го. Число нейронов в каждом слое может быть любым и никак заранее не связано с количеством нейронов в других слоях. Стандартный способ подачи входных сигналов: все нейроны первого слоя получают каждый входной сигнал.

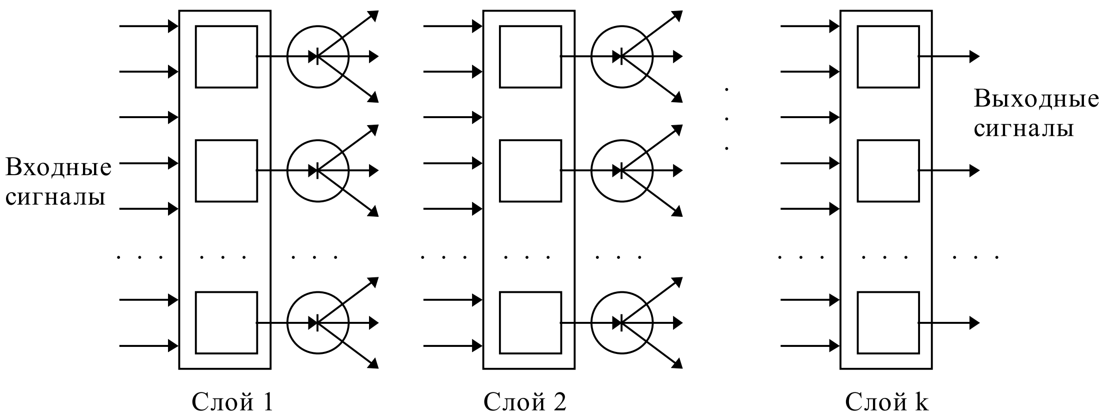

Рисунок 1.7 Многослойная сеть

1.3 Аппроксимирующие свойства произвольной нелинейности

Из предыдущего пункта видно, что искусственные нейронные сети вычисляют линейные функции, нелинейные функции одного переменного, а также всевозможные суперпозиции – функции от функций. Какие функции удастся вычислить точно, а какие функции можно сколь угодно точно аппроксимировать с помощью искусственных нейронных сетей? Для ответа на эти вопросы ниже приведен ряд теорем.

Можно ли произвольную непрерывную функцию n переменных получить с помощью операций сложения, умножения и суперпозиции из непрерывных функций одного переменного? В серии работ А.Н. Колмогоров, а затем В.И. Арнольд решили эту проблему.

Последняя теорема Колмогорова, завершившая серию исследований для непрерывных функций, утверждает: каждая непрерывная функция n переменных, заданная на единичном кубе n-мерного пространства, представима в виде

$$f(x_1, x_2, ..., x_n) = \sum_{q=1}^{2n+1} h_q \left[\sum_{p=1}^{n} \varphi_q^p (x_p) \right], \tag{1.1}$$

где функции $h_q(u)$ непрерывны, а функции $\varphi_q^p(x_p)$, кроме того, еще и стандартны, то есть не зависят от выбора функции f.

Кроме вопроса о точном представлении функций существует еще один вопрос – об аппроксимации. Приближение функций многочленами и рациональными функциями имеет историю, еще более давнюю, чем проблема точного представления.

Теорема Вейерштрасса утверждает, что непрерывную функцию нескольких переменных $f(x_1, x_2, ..., x_n)$ на замкнутом ограниченном множестве Q можно равномерно приблизить последовательностью полиномов: для любого ε>0 существует такой многочлен $P(x_1, x_2, ..., x_n)$, что

$$\sup_{Q} |f(x_1, x_2, ..., x_n) - P(x_1, x_2, ..., x_n)| < \varepsilon \tag{1.2}$$

Чтобы сформулировать обобщения и усиления теоремы Вейерштрасса, необходимо перейти к несколько более абстрактному языку. Рассмотрим компактное пространство X и алгебру $C(X)$ непрерывных функций на X с вещественными значениями.

Теорема Стоуна утверждает: пусть $E \subseteq C(X)$ - замкнутая подалгебра в $C(X)$, $1 \in E$ и функции из E разделяют точки в X (то есть для любых различных $x,y \in X$ существует такая функция $g \in E$, что $g(x) \neq g(y)$). Тогда $E=C(X)$.

Пусть $E \subseteq C(X)$ - линейное пространство, $C(\mathbf{R})$ - пространство непрерывных функций на действительной оси $\mathbf{R}$, $f \in C(\mathbf{R})$ - нелинейная функция и для любого $g \in E$ выполнено $f(g) \in E$. В этом случае будем говорить, что E замкнуто относительно нелинейной унарной операции f.

Обобщенная теорема Стоуна утверждает: пусть $E \subseteq C(X)$ - замкнутое линейное подпространство в $C(X)$, $1 \in E$, функции из E разделяют точки в X и E замкнуто относительно нелинейной унарной операции $f \in C(\mathbf{R})$. Тогда $E=C(X)$.

Рассмотрим множество всех таких $p \in C(\mathbf{R})$, что $p(E) \subseteq E$, то есть для любого $g \in E$ выполнено: $p(g) \in E$. Обозначим это множество P_E. Оно обладает следующими свойствами:

1. P_E - полугруппа относительно суперпозиции функций.
2. P_E - замкнутое линейное подпространство в $C(\mathbf{R})$ (в топологии равномерной сходимости на компактах).
3. $1 \in P_E$ и $\mathrm{id} \in P_E$ ($\mathrm{id}(x) \equiv x$).
4. P_E включает хоть одну непрерывную нелинейную функцию.

Пусть множество $P \subseteq C(\mathbf{R})$ удовлетворяет условиям 1-4. Тогда $P=C(\mathbf{R})$.

Формулировка обобщенной теоремы Стоуна сложна для понимания. Её можно трактовать как утверждение об универсальных аппроксимационных свойствах произвольной нелинейности. С помощью линейных операций и каскадного соединения можно из произвольного нелинейного элемента получить устройство, вычисляющее любую непрерывную функцию с любой наперед заданной точностью.

1.4 Искусственные нейронные сети и непрерывные автоматы

Рассмотрим вопрос об имитации гладких автоматов с помощью нейронных сетей. Если есть возможность с любой точностью приблизить любую непрерывную функцию, и нет ограничений на способы соединения устройств, то можно сколь угодно точно имитировать работу любого непрерывного автомата. Покажем это.

Каждый автомат имеет несколько входов (n), несколько выходов (p) и конечный набор (s) параметров состояния. Он вычисляет $s+p$ функций от $n+s$ переменных. Аргументы этих функций - входные сигналы (их n) и текущие параметры состояния (их s). Значения функций – выходные сигналы (их p) и параметры состояния на следующем шаге (их s). Каждый такой автомат можно представить как систему из $s+p$ более простых автоматов. Эти простые автоматы вычисляют по одной функции от $n+s$ переменных. Смена состояний достигается за счет того, что часть значений этих функций на следующем шаге становится аргументами – так соединены автоматы.

Таким образом, без потери общности можно рассматривать сеть автоматов как набор устройств, каждое из которых вычисляет функцию нескольких переменных $f(x_1, \ldots ,x_n)$. Это позволяет использовать предыдущие результаты. Нейронные сети позволяют с любой точностью вычислять произвольную непрерывную функцию $f(x_1, \ldots ,x_n)$. Следовательно, с их помощью можно сколь угодно точно аппроксимировать функционирование любого непрерывного автомата.

1.5 Требования к алгоритму обучения

Обучение нейронных сетей можно представить как задачу оптимизации в тех случаях, когда удается оценить работу сети. Строится функция оценки.. Простейший и самый распространенный пример оценки - сумма квадратов расстояний от выходных сигналов сети до их требуемых значений:

$$E = \frac{\sum (Z - Z^*)^2}{2} \tag{1.3}$$

где Z и Z^* - полученное и требуемое значение выходного сигнала.

Для обучения используется оценка, усредненная по примерам с известным ответом. Предлагается рассматривать обучение нейронных сетей как задачу оптимизации. Это означает, что весь мощный арсенал методов оптимизации может быть испытан для обучения. Существует, однако, ряд специфических ограничений. Они связаны с огромной размерностью задачи обучения. Число параметров может достигать 10^8 - и даже более. Уже в простейших программных имитаторах на персональных компьютерах подбирается 10^3 - 10^4 параметров.

Из-за высокой размерности возникает два требования к алгоритму:

1. Ограничение по памяти. Пусть n - число параметров. Если алгоритм требует затрат памяти порядка n^2 ,то он вряд ли применим для обучения. Вообще говоря, желательно иметь алгоритмы, которые требуют затрат памяти порядка Kn, K=const.
2. Возможность параллельного выполнения наиболее трудоемких этапов алгоритма и желательно - нейронной сетью.

Еще два обстоятельства связаны с нейрокомпьютерной спецификой:

1. Обученный нейрокомпьютер должен с приемлемой точностью решать все тестовые задачи. Поэтому задача обучения становится по существу многокритериальной задачей оптимизации: надо найти точку общего минимума большого числа функций. Обучение нейрокомпьютера исходит

из гипотезы о существовании такой точки. Основания гипотезы - очень большое число переменных и сходство между функциями. Само понятие «сходство» здесь трудно формализовать, но опыт показывает что предположение о существовании общего минимума или, точнее, точек, где значения всех оценок мало отличаются от минимальных, часто оправдывается.

2. Обученный нейрокомпьютер должен иметь возможность приобретать новые навыки без утраты старых. Возможно более слабое требование: новые навыки могут сопровождаться потерей точности в старых, но эта потеря не должна быть особо существенной, а качественные изменения должны быть исключены. Это означает, что в достаточно большой окрестности найденной точки общего минимума оценок значения этих функций незначительно отличаются от минимальных. Мало того, что должна быть найдена точка общего минимума, так она еще должна лежать в достаточно широкой низменности, где значения всех минимизируемых функций близки к минимуму. Для решения этой задачи нужны специальные средства.

1.6 Алгоритм обратного распространения ошибки

Рассмотрим обучение многослойной искусственной нейронной сети (рис.1.7) с функцией ошибки (1.3) в качестве целевой функции.

Сеть задается вектором параметров – весовыми коэффициентами α .

При заданном обучающем множестве ошибка сети зависит только от вектора параметров:

$$E = E(\alpha) \tag{1.4}$$

При обучении на каждой итерации будем корректировать параметры в направлении антиградиента E:

$$\Delta\alpha = -\varepsilon\nabla E(\alpha) \tag{1.5}$$

В теории оптимизации доказано, что такой алгоритм обеспечивает сходимость к одному из локальных минимумов функции ошибки, при условии правильного выбора $\varepsilon > 0$ на каждой итерации. Такой метод оптимизации называется методом наискорейшего спуска.

Коррекции необходимо рассчитывать на каждой итерации. Поэтому каждая итерация требует расчета компонент градиента ∇E и выбора оптимального шага ε. Как рассчитать градиент ∇E с наименьшими вычислительными затратами?

Самое простое, но не самое лучшее решение для этого – воспользоваться стандартным определением градиента:

$$\frac{\partial E(\alpha)}{\partial \alpha_i} = \lim_{\Delta\alpha_i \to 0} \frac{E(\alpha + \Delta\alpha_i) - E(\alpha)}{\Delta\alpha_i} \tag{1.6}$$

где $\Delta\alpha_i$ - приращение i-й компоненты вектора параметров α.

Однако, чтобы рассчитать каждое значение функции E, требуется подать входной вектор и просчитать выходные значения для всех нейронов в сети. Это очень большой объем вычислений. Если учесть, что требуется рассчитать все компоненты градиента таким образом, то неэффективность метода становится очевидной.

Алгоритм обратного распространения – способ ускоренного расчета компонент градиента. Идея метода состоит в том, чтобы представить E в виде сложной функции и последовательно рассчитать частные производные по формуле для сложной функции.

Запишем (1.5) для весовых коэффициентов:

$$\Delta\alpha_{ijk} = -\varepsilon \cdot \left. \left(\frac{\partial E(\alpha)}{\partial \alpha_{ijk}} \right) \right|_{\alpha} \tag{1.7}$$

где i - номер слоя, j - номер нейрона в слое, k – номер входа нейрона.

Для слоя легко записать компоненты градиента по весам как производную сложной функции:

$$\frac{\partial E}{\partial \alpha_{ijk}} = \frac{\partial E}{\partial \varphi\left(\sum_k \alpha_{ijk} \cdot x_{ijk}\right)} \cdot \frac{\partial \varphi\left(\sum_k \alpha_{ijk} \cdot x_{ijk}\right)}{\partial \sum_k \alpha_{ijk} \cdot x_{ijk}} \cdot \frac{\partial \sum_k \alpha_{ijk} \cdot x_{ijk}}{\partial \alpha_{ijk}} \tag{1.8}$$

Для выходного слоя:

$$\frac{\partial E}{\partial \varphi\left(\sum_k \alpha_{ijk} \cdot x_{ijk}\right)} = \sum \left(Z - Z^*\right) \tag{1.9}$$

Для предыдущих слоев:

$$\frac{\partial E}{\partial \varphi\left(\sum_k \alpha_{ijk} \cdot x_{ijk}\right)} = \frac{\partial E}{\partial x_{i+1jk}} \tag{1.10}$$

Производная от функции активации при $\phi(x) = th(x)$:

$$\frac{\partial \varphi\left(\sum_k \alpha_{ijk} \cdot x_{ijk}\right)}{\partial \sum_k \alpha_{ijk} \cdot x_{ijk}} = 1 - \phi^2\left(\sum_k \alpha_{ijk} \cdot x_{ijk}\right) \tag{1.11}$$

Производная от функции активации при $\phi(x) = \dfrac{1}{1+e^{-x}}$:

$$\frac{\partial \varphi\left(\sum_k \alpha_{ijk} \cdot x_{ijk}\right)}{\partial \sum_k \alpha_{ijk} \cdot x_{ijk}} = \phi\left(\sum_k \alpha_{ijk} \cdot x_{ijk}\right) \cdot \left(1 - \phi\left(\sum_k \alpha_{ijk} \cdot x_{ijk}\right)\right) \tag{1.12}$$

Производная от взвешенной суммы по весам:

$$\frac{\partial \sum_k \alpha_{ijk} \cdot x_{ijk}}{\partial \alpha_{ijk}} = x_{ijk} \tag{1.13}$$

Мы получили полный набор формул обратного распространения, который дает компонент градиента для всех слоев и всех нейронов в сети. Зная градиент, можно проводить обучение в иде итераций по формуле (1.5).

Обратное распространение ошибки позволяет во много раз сократить вычислительные затраты на расчет градиента по сравнению с расчетом градиента по определению. Быстрый расчет градиента необходим во многих методах оптимизации(обучения), поэтому значение алгоритма обратного распространения в нейроинформатике велико.

2 НЕЙРОСЕТЕВЫЕ МОДЕЛИ СЛОЖНЫХ СИСТЕМ

2.1 Сложные системы

Рассмотрим систему, состоящую из некоторого числа компонент. Нашей конечной целью будет построение модели системы, описывающей ее поведение, и обладающей предсказательными свойствами. Модель способна во многих приложениях заменить собой исследуемую систему.

Каждая из компонент системы имеет свои свойства и характер поведения в зависимости от собственного состояния и внешних условий. Если все возможные проявления системы сводятся к сумме проявлений ее компонент, то такая система является простой. Для описания простых систем традиционно применяются методы анализа, состоящие в последовательном расчленении системы на компоненты и построении моделей все более простых элементов. Таковым в своей основе является метод математического моделирования, в котором модели описываются в форме уравнений, а предсказание поведения системы основывается на их решении.

Современные технические системы приближаются к такому уровню сложности, когда их наблюдаемое поведение и свойства не сводятся к простой сумме свойств отдельных компонент. При объединении компонент в систему возникают качественно новые свойства, которые не могут быть установлены посредством анализа свойств компонент.

Такие системы, в которых при вычленении компонент могут быть потеряны принципиальные свойства, а при добавлении компонент возникают качественно новые свойства, будем называть сложными. Модель сложной системы, основанная на принципах анализа, будет неустранимо неадекватной изучаемой системе, поскольку при разбиении системы на составляющие ее компоненты теряются ее качественные особенности.

2.2 Принципы информационного кибернетического моделирования

Возможным выходом из положения является построение модели на основе синтеза компонент. В последнее время синтетические информационные модели широко используются и при изучении технических и инженерных систем. В ряде приложений информационные и математические компоненты могут составлять единую модель (например, внешние условия описываются решениями уравнений математической физики, а отклик системы - информационной моделью).

Основным принципом информационного моделирования является принцип «черного ящика». В противоположность аналитическому подходу, при котором моделируется внутренняя структура системы, в синтетическом методе «черного ящика» моделируется внешнее функционирование системы. С точки зрения пользователя модели структура системы спрятана в черном ящике, который имитирует поведенческие особенности системы.

Кибернетический принцип «черного ящика» был предложен в рамках теории идентификации систем, в которой для построения модели системы предлагается широкий параметрический класс базисных функций или уравнений, а сама модель синтезируется путем выбора параметров из условия наилучшего, при заданной функции ценности, соответствия решений уравнений поведению системы. При этом структура системы никак не отражается в структуре уравнений модели.

Функционирование системы в рамках синтетической модели описывается чисто информационно, на основе данных экспериментов или наблюдений над реальной системой. Как правило, информационные модели проигрывают формальным математическим моделям и экспертным системам по степени «объяснимости» выдаваемых результатов, однако отсутствие ограничений на сложность моделируемых систем определяет их важную практическую значимость.

2.3 Типы информационных моделей

Можно выделить несколько типов информационных моделей, отличающихся по характеру запросов к ним. Перечислим лишь некоторые из них:

1. Моделирование отклика системы на внешнее воздействие.
2. Классификация внутренних состояний системы.
3. Прогноз динамики изменения системы.
4. Оценка полноты описания системы и сравнительная информационная значимость параметров системы.
5. Оптимизация параметров системы по отношению к заданной функции ценности.
6. Адаптивное управление системой.

В этом разделе изложение будет основываться на моделях первого из указанных типов.

Пусть X - вектор, компоненты которого соответствуют количественным свойствам системы, X' - вектор количественных свойств внешних воздействий. Отклик системы может быть описан некоторой (неизвестной) вектор-функцией F: $Y = F(X,X')$, где Y - вектор отклика. Задачей моделирования является идентификация системы, состоящая в нахождении функционального отношения, алгоритма или системы правил в общей форме $Z=G(X,X')$, ассоциирующей каждую пару векторов (X,X') с вектором Z таким образом, что Z и Y близки в некоторой метрике, отражающей цели моделирования. Отношение $Z=G(X,X')$, воспроизводящее в указанном смысле функционирование системы F, будем называть информационной моделью системы F.

2.4 Искусственные нейронные сети в информационном моделировании

Искусственные нейронные сети являются удобным и естественным базисом для представления информационных моделей. ИНС выполняет функциональное соответствие между входом и выходом, и может служить информационной моделью G системы F.

Определяемая ИНС функция может быть произвольной при легко выполнимых требованиях к структурной сложности сети и наличии нелинейности в переходных функциях нейронов. Возможность представления любой системной функции F с наперед заданной точностью определяет нейросеть, как компьютер общего назначения. Этот компьютер, в сравнении с машиной фон Неймана, имеет принципиально другой способ организации вычислительного процесса - он не программируется с использованием явных правил и кодов в соответствии с заданным алгоритмом, а обучается посредством целевой адаптации синаптических связей для представления требуемой функции.

В гипотетической ситуации, когда функция системы F известна или известен алгоритм ее вычисления при произвольных значениях аргументов, машина фон Неймана наилучшим средством для моделирования (состоящего в вычислении F), и необходимость в информационных моделях отпадает.

При моделировании реальных сложных технических систем значения системной функции F получаются на основе экспериментов или наблюдений, которые проводятся лишь для конечного числа параметров X. При этом значения как Y так и X измеряются приближенно, и подвержены ошибкам различной природы. Целью моделирования является получение значений системных откликов при произвольном изменении X. В этой ситуации может быть успешно применена информационная (статистическая) модель G исследуемой системы F.

2.5 Характер приближений в информационных моделях

Специфичность информационных моделей проявляется не только в способах их синтеза, но и характере делаемых приближений. Отличия в поведении системы и ее информационной модели возникают вследствие свойств экспериментальных данных:

1. Информационные модели являются неполными. Пространства входных и выходных переменных не могут, в общем случае, содержать все параметры, существенные для описания поведения системы. Это связано как с техническими ограничениями, так и с ограниченностью наших представлений о моделируемой системе. Кроме того, при увеличении числа переменных ожесточаются требования на объем необходимых экспериментальных данных для построения модели. Эффект опущенных (скрытых) входных параметров может нарушать однозначность моделируемой системной функции F.
2. База экспериментальных данных, на которых основывается модель G, рассматривается как внешняя данность. При этом, в данных всегда присутствуют ошибки разной природы, шум, а также противоречия отдельных измерений друг другу. За исключением простых случаев, искажения в данных не могут быть устранены полностью.
3. Экспериментальные данные, как правило, имеют произвольное распределение в пространстве переменных задачи. Как следствие, получаемые модели будут обладать неодинаковой достоверностью и точностью в различных областях изменения параметров.
4. Экспериментальные данные могут содержать пропущенные значения (например, вследствие потери информации, отказа измеряющих датчиков, невозможности проведения полного набора анализов и т.п.). Произвольность в интерпретации этих значений, опять-таки, ухудшает свойства модели.

Такие особенности в данных и в постановке задач требуют особого отношения к ошибкам информационных моделей.

2.6 Ошибка обучения и ошибка обобщения

Итак, при информационном подходе требуемая модель G системы F не может быть полностью основана на явных правилах и формальных законах. Процесс получения G из имеющихся отрывочных экспериментальных сведений о системе F может рассматриваться, как обучение модели G поведению F в соответствии с заданным критерием, настолько близко, насколько возможно. Алгоритмически, обучение означает подстройку внутренних параметров модели с целью минимизации ошибки модели:

$$E = \| G - F \| \tag{2.1}$$

Прямое измерение указанной ошибки модели на практике не достижимо, поскольку системная функция F при произвольных значениях аргумента не известна. Однако возможно получение ее оценки:

$$E_L = \sum_{X \in \mathbf{X}} \| G(X) - Y \| \tag{2.2}$$

где суммирование по X проводится по некоторому конечному набору параметров $\boldsymbol{X}$, называемому обучающим множеством. При использовании базы данных наблюдений за системой, для обучения может отводиться некоторая ее часть, называемая в этом случае обучающей выборкой. Для обучающих примеров X отклики системы Y известны. Норма невязки модельной функции G и системной функции Y на множестве $\boldsymbol{X}$ играет важную роль в информационном моделировании и называется ошибкой обучения модели.

В приложениях пользователя обычно интересуют предсказательные свойства модели. При этом главным является вопрос, каковым будет отклик системы на новое воздействие, пример которого отсутствует в базе данных наблюдений. Наиболее общий ответ на этот вопрос дает ошибка модели E. Неизвестная ошибка, допускаемая моделью G на данных, не использовавшихся при обучении, называется ошибкой обобщения модели E_G.

Основной целью при построении информационной модели является уменьшение именно ошибки обобщения, поскольку малая ошибка обучения гарантирует адекватность модели лишь в заранее выбранных точках (а в них значения отклика системы известны и без всякой модели).

Важно отметить, что малость ошибки обучения не гарантирует малость ошибки обобщения. Поскольку истинное значение ошибки обобщения не доступно, в практике используется ее оценка. Для ее получения анализируется часть примеров из имеющейся базы данных, для которых известны отклики системы, но которые не использовались при обучении. Эта выборка примеров называется тестовой выборкой. Ошибка обобщения оценивается, как норма уклонения модели на множестве примеров из тестовой выборки.

2.7 Типы задач информационного моделирования

При формулировании задачи предсказания реакции исследуемой системы при ее известном состоянии на заданные внешние воздействия, т.е. получения величин Y при заданных X исследователь имеет дело с прямой задачей. Другим важным классом информационных задач являются обратные задачи. Целью обратной задачи выступает получение входных величин X, соответствующих наблюдаемым значениям выходов Y. При моделировании сложных систем соответствующий запрос к модели формулируется, как поиск внешних условий, которые привели к реализовавшемуся отклику системы.

Для большинства приложений чисто обратные задачи встречаются относительно редко, так как обычно имеются дополнительные сведения о системе. Например, кроме измеренного отклика, могут быть известны переменные состояния системы и часть параметров воздействия. В этом случае задача относится к классу комбинированных задач: по известным значениям части компонент входного X и выходного Y векторов восстановить оставшиеся неизвестные компоненты.

В общем случае моделируемая системная функция может быть представлена в виде $(X,Y)=\boldsymbol{F}\,(X, Y)$. В этом случае комбинированный вектор (X,Y) рассматривается одновременно, как входной и выходной. В этом смысле, произвольная задача допускает комбинированную постановку.

2.8 Некорректность задач информационного моделирования

Отличительная особенность обратных и комбинированных задач состоит в том, что они обычно являются некорректно поставленными, и поэтому требуют специализированных методов поиска приближенных решений. Согласно Ж.Адамару, для корректности постановки задачи необходимо:

1. Существование решения при всех допустимых исходных данных.
2. Единственность данного решения.
3. Устойчивость решения к малым изменениям исходных данных.

Рассмотрим характер возможных нарушений данных условий при решении модельной обратной задачи.

Пусть имеется три исследуемых систем, описываемых кусочно-линейными функциями одной переменной $y=F(x)$ на отрезке [0..1]. Системы отличаются друг от друга величиной скачка h системной функции (рис.2.1). Прямая задача состоит в построении приближения G к функции F, с использованием пар значений $\{x_i,\ y_i=s(x_i)\}$, где x_i - конечный набор N_α случайных равномерно распределенных на [0..1] точек. Обратная задача заключается в нахождении функции, аппроксимирующей соотношения $x_i(y_i)$.

В зависимости от величины скачка моделируемой функции можно выделить три варианта.

Система А (h=0). Модель является линейной: $y=x$. Для прямой задачи легко получить исчезающую ошибку обучения $E_L \approx 0$, и малую ошибку обобщения E_G. Для обратной задачи получаются такие же результаты, так она при точных значениях $\{x_i, y_i\}$ не содержит некорректности. Задачи с корректными решениями на всей области определения и множестве значений – безусловно корректные. Корректность постановки обратной задачи для системы А определяется существованием однозначной и непрерывной функции F^{-1}.

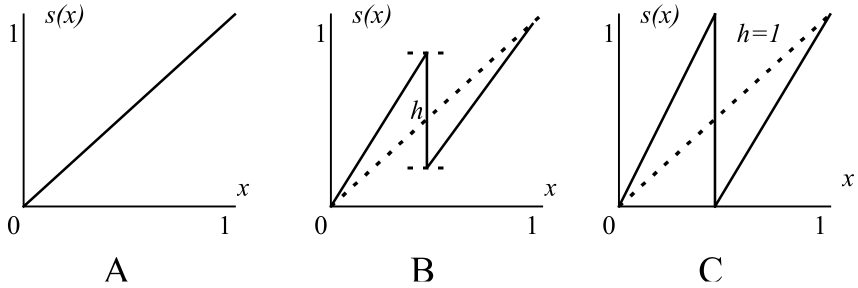

Рисунок 2.1 Модельные системы с различными величинами скачка системной функции.

Система В (0<h<1). Прямая задача в этом случае также хорошо определена, и при использовании достаточно богатого множества базисных функций можно произвольно уменьшить ошибку обучения ($E_L \approx 0$) при хорошем обобщении. Обратная задача характеризуется наличием на множестве значений областей с однозначной ($y > 0.5+0.5h$; $y < 0.5-0.5h$) и неоднозначной ($y \in [0.5-0.5h, 0.5+0.5h]$) обратной функцией. В областях однозначности функции могут быть получены произвольно точные результаты для обратной задачи. Однако в отрезке нарушения однозначности ошибка обучения (и ошибка обобщения) останется конечной, поскольку противоречие в данных, полученных из разных ветвей обратной функции, не устранимо. Значение ошибки обобщения пропорционально длине отрезка неоднозначности h. Такие задачи, корректное (единственное и устойчивое) решение которых может быть получено только для некоторой подобласти множества значений, будем называть условно (или частично) корректными.

Система С (h=1). Прямая задача по-прежнему корректно поставлена, требуемое обучение и обобщение может быть достигнуто ($E_L \approx 0$). Однако ситуация качественно меняется для случая обратной задачи. Обратная функция двузначна на всем множестве значений, информация о ее значении минимальна. Обратная задача полностью некорректно поставлена.

Заметим, что прямая задача является безусловно корректной только при полном отсутствии шума в обучающих данных. При наличии случайных компонент в значениях X имеется целое «облако» решений прямой задачи, причем размер облака пропорционален величине шума. Таким образом, нарушается единственность решения прямой задачи, и она становится некорректно поставленной.

ВЫВОДЫ

Искусственные нейронные сети предоставляют универсальные возможности для моделирования сложных систем, так как позволяют реализовывать любые функции действительных переменных. Искусственные нейронные сети реализуют свою вычислительную мощь, благодаря двум основным своим свойствам: существенно параллельной распределенной структуре и способности обучаться и обобщать полученные знания. Эти свойства делают ИНС универсальной системой переработки информации, которая решает сложные задачи управления, непосильные другим техникам.

ПЕРЕЧЕНЬ ССЫЛОК

1. Юревич Е.А. Теория автоматического управления. – Ленинград: Энергия, 1975. – 416 с.
2. Заенцев И.В. Нейронные сети: основные модели – Воронеж 1999. – 76 с.
3. Миркес Е.М. Учебное пособие по курсу нейроинформатика – Красноярск 2002. – 347 с.
4. Горбань А.Н., Дудин-Барковский В.Л. Нейроинформатика. – Новосибирск: Наука. Сибирское предприятие РАН, 1998. – 296с.
5. Вороновский Г.К., Махотило К.В. Генетические алгоритмы, искусственные нейронные сети и проблемы виртуальной реальности – Харьков: Основа, 1997. – 107 с.
6. Уоссермен Ф. Нейрокомпьютерная техника: теория и практика – 1992.