На правах рукописи

Вичугов Владимир Николаевич

НЕЙРОСЕТЕВОЙ МЕТОД УПРАВЛЕНИЯ НА ОСНОВЕ ПОДКРЕПЛЯЕМОГО ОБУЧЕНИЯ

Специальность 05.13.01

Системный анализ, управление и обработка информации

(отрасль: промышленность)

АВТОРЕФЕРАТ

диссертации на соискание ученой степени

кандидата технических наук

Томск – 2008

Работа выполнена в Государственном образовательном учреждении высшего профессионального образования «Томский политехнический университет»

Научный руководитель:	доктор технических наук, профессор Цапко Геннадий Павлович
Официальные оппоненты:	доктор технических наук, с.н.с. Калайда Владимир Тимофеевич
	кандидат технических наук, доцент Тараканов Дмитрий Викторович
Ведущая организация:	Санкт-Петербургский государственный электротехнический университет «ЛЭТИ», г. Санкт-Петербург

Защита состоится 10 декабря 2008 г. в 14 ^{30} на заседании совета по защите докторских и кандидатских диссертаций Д 212.269.06 при Томском политехническом университете по адресу: 634034, г. Томск, ул. Советская, 84, институт «Кибернетический Центр» ТПУ.

С диссертацией можно ознакомиться в Научно-технической библиотеке Томского политехнического университета по адресу: 634034, г. Томск, ул. Белинского, 55.

Автореферат разослан 7 ноября 2008 г.

Ученый секретарь Совета
кандидат технических наук, доцент М.А. Сонькин

Общая характеристика работы

Актуальность исследования. Постоянное усложнение технических объектов управления (ОУ) и расширение областей их применения приводит к необходимости развития средств и методов интеллектуального управления в условиях неопределенности и при изменяющихся условиях функционирования. Применение методов классической теории автоматического управления для управления сложными динамическими ОУ затруднено рядом факторов. Прежде всего, это сложность получения достаточно точного формализованного описания ОУ. Кроме того, параметры ОУ могут изменяться в широких пределах в процессе функционирования системы, либо иметь большой разброс значений от образца к образцу. Также следует учесть, что практически все реальные ОУ являются нелинейными, и их представление в виде линейных математических моделей является лишь приблизительным. Кроме того, наличие в реальных сигналах помех вносит дополнительные трудности в процесс получения адекватного математического описания ОУ. Преодоление указанных трудностей связывают с развитием интеллектуальных систем управления, основанных, в частности, на применении аппарата искусственных нейронных сетей.

Начиная с 1990-х гг. активно развивается метод подкрепляемого обучения (англ. reinforcement learning), относящийся к группе методов машинного обучения. В основе этого метода лежат те основополагающие принципы адаптивного поведения, которые позволяют живым организмам приспосабливаться к изменяющимся или неизвестным условиям обитания. В этом методе рассматривается система, которая в процессе взаимодействия с внешней средой получает сигнал подкрепления, характеризующий, насколько хорошо функционирует система в текущий момент времени. Алгоритмы, относящиеся к методу подкрепляемого обучения, определяют порядок изменения состояния системы таким образом, чтобы формируемые воздействия системы на внешнюю среду обеспечивали максимальное значение суммарного сигнала подкрепления, накопленного за длительный период времени. Одной из отличительных особенностей метода подкрепляемого обучения является тот факт, что в начале функционирования система не обладает практически никакой информацией о внешней среде, и обучение системы происходит в процессе взаимодействия с ней. Второй особенностью метода подкрепляемого обучения является формирование воздействий с учетом сигналов подкрепления, которые будут получены в отдаленном будущем.

Целью работы является разработка нейросетевого метода адаптивного управления, основанного на принципах подкрепляемого обучения и обеспечивающего формирование управляющих воздействий на основе взаимодействия с объектом управления.

Для достижения поставленной цели были решены следующие **задачи**:

1. Разработка модифицированного градиентного алгоритма обучения радиально-бизисных нейронных сетей (РБНС), обеспечивающего динамическое изменение структуры нейронной сети в процессе обучения.

2. Разработка обобщенной структурной схемы нейросетевой RL-САУ и алгоритмов работы структурных блоков.

3. Разработка программного средства для моделирования нейросетевой RL-САУ.

4. Определение рекомендаций по настройке параметров управляющего устройства (УУ) в процессе работы RL-САУ.

5. Апробация разработанного метода управления в задачах управления линейными и нелинейными ОУ.

Методы исследований. В работе использованы методы теории управления, теории оптимизации, системного анализа, математического моделирования, прикладной математики и теории нейронных сетей.

Научную новизну работы определяют:

1. Модифицированный градиентный алгоритм обучения РБНС, отличающийся от классического градиентного алгоритма возможностью динамического изменения структуры РБНС.

2. Нейросетевой метод адаптивного управления, основанный на разработанной обобщенной структурной схеме нейросетевой RL-САУ и алгоритмах функционирования структурных блоков и обеспечивающий формирование управляющих воздействий на основе взаимодействия с ОУ.

3. Алгоритм адаптивного изменения значения параметра обучения в процессе функционирования нейросетевой RL-САУ, обеспечивающий устойчивость процесса обучения при использовании РБНС для представления функции оценки воздействия.

Практическая значимость и реализация результатов работы.

Разработанный метод управления может быть использован при разработке адаптивных систем управления, когда отсутствует априорная информация о математической модели ОУ. Разработанное программное средство моделирования нейросетевой RL-САУ может быть использовано для определения последовательности управляющих воздействий, переводящих ОУ из начального состояния в требуемое.

Разработанное программное средство моделирования нейросетевой RL-САУ используется в ОАО «Информационные спутниковые системы» имени академика М.Ф. Решетнева». Результат внедрения подтвержден соответствующим актом.

Основные положения, выносимые на защиту:

1. Разработанный модифицированный градиентный алгоритм обучения РБНС обеспечивает автоматическое формирование структуры нейронной сети в процессе обучения.

2. Нейросетевая RL-САУ позволяет формировать управляющие воздействия на ОУ в соответствии с выбранным критерием функционирования системы при неизвестных или изменяющихся свойствах ОУ.

3. Разработанный алгоритм адаптивного изменения значения параметра обучения позволяет обеспечить устойчивость процесса обучения при использовании РБНС для представления функции оценки воздействия.

Публикации. По результатам исследований опубликовано 14 работ, из них одна работа в издании, рекомендуемом списком ВАК.

Структура и объем работы. Диссертационная работа состоит из введения, трех глав, заключения, двух приложений. Основной текст изложен на 140 страницах, общий объем работы – 148 страниц. Диссертация включает 64 рисунка, 5 таблиц. Список использованных источников содержит 92 наименования.

Основное содержание работы

Во **введении** обоснована актуальность диссертационной работы, сформулированы цель и задачи исследования.

В **первой главе** приведено описание метода подкрепляемого обучения, проведен анализ алгоритмов обучения.

В методе подкрепляемого обучения рассматривается агент, взаимодействующий с внешней средой в дискретные моменты времени t_i, называемые тактами (рисунок 1). В данной работе сохранена терминология, которую использовали авторы метода подкрепляемого обучения. Под агентом понимается некоторая автономная система, которая имеет возможность получать информацию о состоянии внешней среды и формировать воздействия, которые приводят к изменению состояния внешней среды. Внешней средой называется все, что находится вне агента и с чем он взаимодействует.

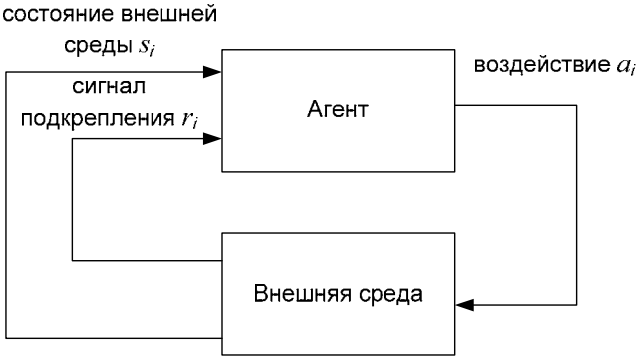

Рисунок 1 – Схема взаимодействия агента со средой

В каждый такт t_i агент получает информацию о состоянии внешней среды $s_i \in S$, где S – конечное множество возможных состояний внешней среды, и на основе этой информации вырабатывает некоторое воздействие на внешнюю среду $a_i \in A(s_i)$, где $A(s_i)$ – конечное множество воздействий, которые агент может выработать при состоянии внешней среды s_i. На следующем такте воздействие a_i переводит внешнюю среду в новое состояние s_{i+1}. На каждом такте агент получает сигнал подкрепления r_i, который является скалярной величиной и характеризует, насколько хорошо агент функционирует во внешней среде. Целью функционирования агента является максимизация суммарной величины подкрепления R, которая на i-ом такте определяется по выражению

$$R_i = r_{i+1} + \gamma \cdot r_{i+2} + \gamma^2 \cdot r_{i+3} + \ldots = \sum_{k=0}^{\infty} \gamma^k \cdot r_{i+1+k} ,$$

где $\gamma \in [0, 1]$ – параметр дисконтирования сигнала подкрепления, обеспечивающий сходимость суммарной величины подкрепления. Для достижения цели функционирования осуществляется определение значений функции оценки воздействия Q, аргументами которой являются состояние внешней среды s и воздействие a, а значением функции является величина суммарной величины подкрепления для будущих тактов при условии, что на текущем такте при состоянии внешней среды s агент выберет воздействие a:

$$Q(s,a) = R_i = \sum_{k=0}^{\infty} \gamma^k \cdot r_{i+1+k} ,$$

при условии, что $a_i=a$ и $s_i=s$.

В том случае, когда значения функции оценки воздействия для всех возможных значений состояний и воздействий определены, функционирование агента для достижения цели функционирования заключается в выборе воздействия, соответствующего максимальному значению функции оценки воздействия при данном состоянии внешней среды:

$$a_i = \arg\max_{a \in A} Q(s_i, a).$$

В начале функционирования во внешней среде функция оценки воздействия имеет нулевые значения для всех значений аргументов. На каждом такте осуществляется изменение функции оценки воздействия в соответствии с одним из алгоритмов метода подкрепляемого обучения. Проведенный анализ алгоритмов обучения показал, что в том случае, когда функция оценки воздействия представлена с помощью матрицы чисел, предпочтительным является использование алгоритма TD(λ), так как он позволяет за один шаг обучения уточнить значение функции оценки воздействия сразу в нескольких точках. При использовании функционального аппроксиматора для представления функции оценки воздействия наиболее предпочтительным алгоритмом обучения является алгоритм Q-обучения.

Во **второй главе** представлена обобщенная структурная схема RL-САУ, в которой для представления функции оценки воздействия используется матрица вещественных чисел. Входящий в состав RL-САУ ОУ должен удовлетворять следующим условиям:

1. ОУ является одномерным.

2. В любой момент времени можно измерить вектор переменных состояния ОУ. Под переменными состояния ОУ в данной работе подразумевается набор сигналов, который вместе с управляющим воздействием u однозначно определяет значение выходной величины y в будущие моменты времени.

Разработанная структурная схема RL-САУ показана на рисунке 2. Вектор входных сигналов УУ состоит из задающего воздействия g, скорости изменения задающего воздействия g', выходной величины y и вектора переменных состояния ОУ $\overline{X}$. В результате обработки вектора входных сигналов УУ формирует управляющее воздействие u, значение которого является одним из элементов заранее определенного множества возможных воздействий A. Под действием управляющего воздействия u ОУ изменяет свое состояние. Наличие

в векторе входных сигналов производной входного воздействия g' и вектора переменных состояния ОУ $\overline{X}$ обусловлено тем, что в соответствии с методом подкрепляемого обучения сигналы подкрепления и состояния внешней среды должны обладать свойством марковости.

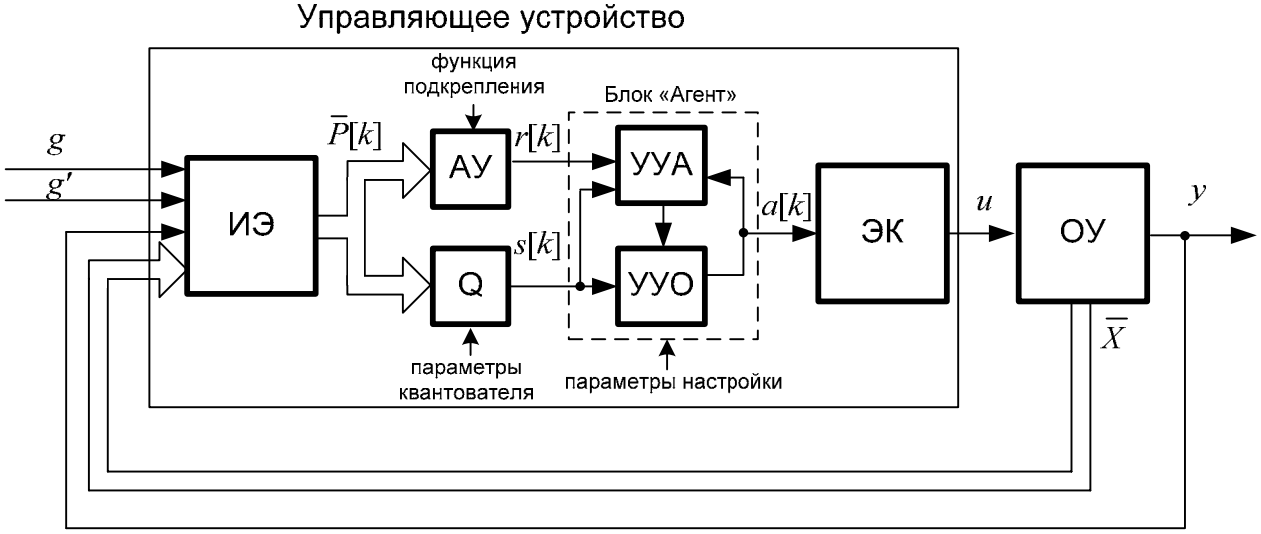

Рисунок 2 – Обобщенная структурная схема RL-САУ

Вектор входных сигналов поступает на вход импульсного элемента (ИЭ), который осуществляет дискретизацию по времени входных сигналов. На выходе ИЭ формируется вектор дискретных сигналов $\overline{P}[k]$, который поступает на анализирующее устройство (АУ) и на квантователь Q. АУ вычисляет значение сигнала подкрепления $r[k]$, а квантователь осуществляет квантование по уровню вектора дискретных сигналов $\overline{P}[k]$ и определяет значение сигнала состояния внешней среды $s[k]$, которое является одним из элементов заранее определенного множества возможных состояний внешней среды S. Устройство управления объектом (УУО) формирует сигнал воздействия на ОУ $a[k]$, а устройство управления адаптацией (УУА) осуществляет коррекцию функции оценки воздействия в соответствии с алгоритмом TD(λ) метода подкрепляемого обучения. Экстраполятор (ЭК) переводит дискретную величину $a[k]$ в непрерывное по времени управляющее воздействие u.

Разработаны алгоритмы работы каждого из элементов обобщенной структурной схемы RL-САУ. Для определения параметров квантователя необходимо задать количество уровней квантования для каждого элемента вектора сигналов $\overline{P}[k]$. Если RL-САУ предназначена для установления выходного сигнала ОУ y, равного задающему воздействию g, то для определения сигнала подкрепления $r[k]$ предлагается использовать выражение

$$r[k] = -(y[k-1] - g[k-1])^2 .$$

Максимальное значение этого выражения равно нулю и достигается только в том случае, когда выходной сигнал ОУ y равен задающему воздействию g.

На основе обобщенной структурной схемы RL-САУ было разработано программное средство «Исследование RL-САУ», главное окно которого показано на рисунке 3.

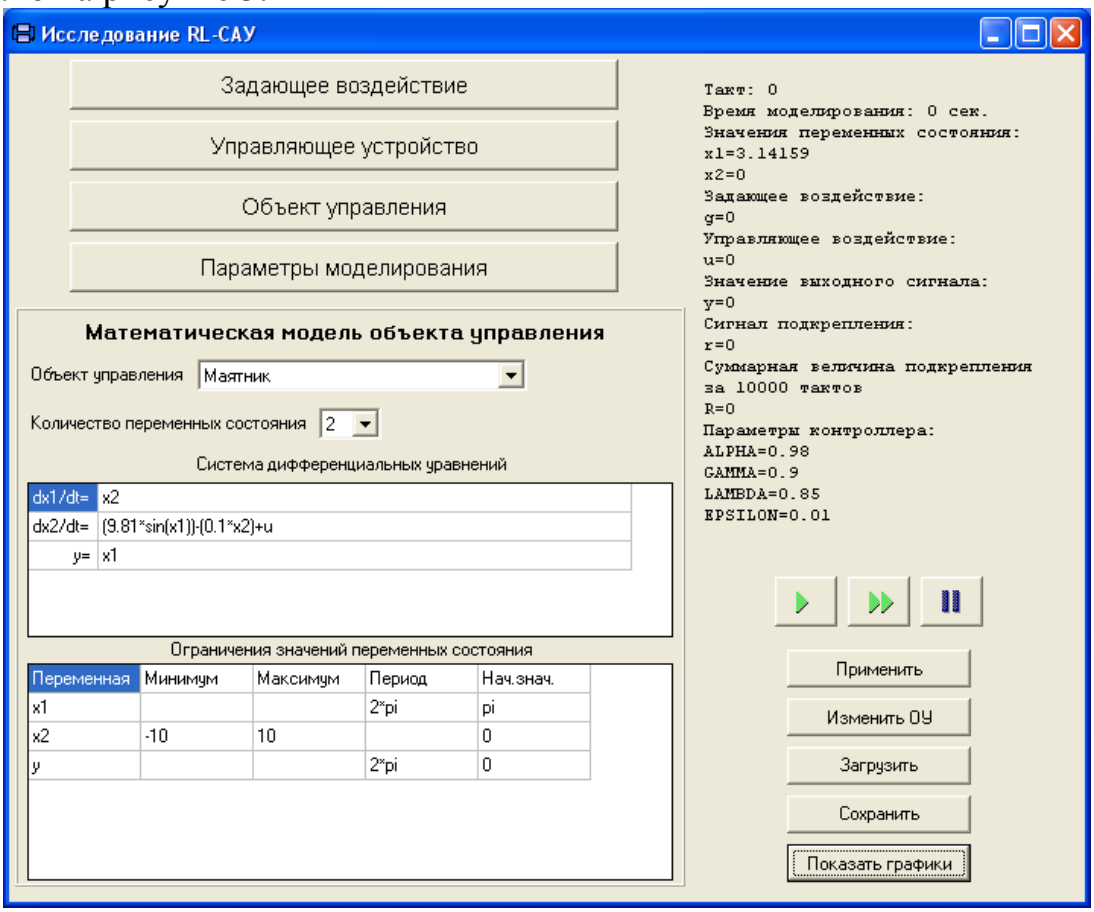

Рисунок 3 – Главное окно программы «Исследование RL-САУ»

Программное средство позволяет задавать математическую модель ОУ в виде системы дифференциальных уравнений, определять вид и параметры задающего воздействия, задавать параметры настройки УУ, управлять процессом моделирования, отображать на экране значения всех моделируемых сигналов и их графики, определять значения показателей качества управления, сохранять результаты моделирования в файлы. Параметры настройки УУ включают в себя параметры алгоритма обучения TD(λ), количество уровней квантования для каждого входного сигнала, а также элементы множества возможных управляющих сигналов.

В программном средстве «Исследование RL-САУ» были проведены экспериментальные исследования систем управления линейными и нелинейными ОУ. Эксперименты показали необходимость перехода к представлению функции оценки воздействия с помощью функционального аппроксиматора вместо использования матрицы вещественных чисел. Данная необходимость определяется экспоненциальным ростом требуемой памяти для хранения матрицы чисел при увеличении порядка ОУ, либо при увеличении количества уровней квантования входных сигналов.

В **третьей главе** представлен нейросетевой метод управления, основанный на структурной схеме нейросетевой RL-САУ, в которой функция

оценки воздействия представлена с помощью радиально-базисной нейронной сети.

Проведенный анализ возможности использования различных типов искусственных нейронных сетей (ИНС) показал, что дополнительное обучение многослойного перцептрона в некотором участке рабочей области приводит к потере обученного состояния во всей рабочей области ИНС, что не позволяет использовать этот тип ИНС для аппроксимации функции оценки воздействия. Указанный недостаток отсутствует в РБНС, так как каждый элемент РБНС влияет на значение выходного сигнала преимущественно только в ограниченном участке рабочей области, который характеризуется положением центра элемента и параметром σ, называемым шириной радиальной функции. Чем больше значение параметра σ, тем больше размер области, на которую оказывает влияние данный элемент. Структура РБНС показана на рисунке 4. РБНС состоит из двух слоев. Все входные сигналы поступают на все элементы первого слоя без изменений.

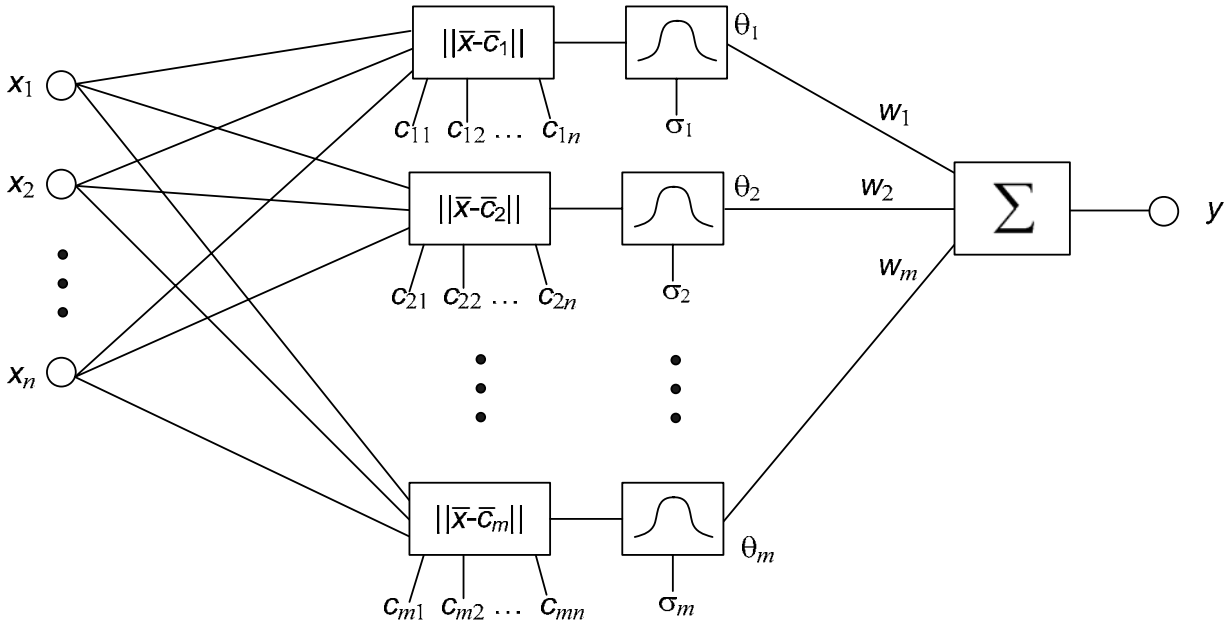

Рисунок 4 – Структура радиально-базисной нейронной сети

Выходной сигнал каждого элемента определяется функцией Гаусса

$$\theta_i = \exp\left(-\frac{\sum_{j=1}^{n}(x_j - c_{ij})^2}{2\sigma_i^2}\right),$$

где σ_i – ширина функции i-го элемента; c_{i1}, c_{i2}, …, c_{in} – координаты центра i-го элемента. Выходной сигнал РБНС вычисляется как взвешенная сумма сигналов элементов:

$$y = \sum_{i=1}^{m} w_i \cdot \theta_i,$$

где w_i – весовой коэффициент выходной связи i-го элемента; m – количество элементов РБНС.

Для обучения РБНС используется градиентный алгоритм, основанный на минимизации целевой функции ошибки РБНС. В соответствии с этим алгоритмом для каждого элемента вычисляется величина изменений весового коэффициента Δw_i, величина изменения ширины элемента $\Delta\sigma_i$ и величины изменения координат центра элемента Δc_{ij}.

В результате проведенных экспериментов, были выявлены некоторые недостатки РБНС:

1. В алгоритме обучения РБНС нет правил для первоначального задания количества элементов сети и их параметров, а так же нет правил для изменения количества элементов в процессе обучения. Равномерное распределение элементов в рабочей области не всегда является оптимальным. Также может возникнуть ситуация, когда количество элементов, заданное первоначально, является недостаточным для достижения требуемого качества обучения.

2. В процессе обучения изменяются параметры всех элементов сети. В результате при увеличении количества элементов вычислительные затраты на обучение также увеличиваются.

3. РБНС не может достичь устойчивого состояния в процессе обучения в тех случаях, когда существуют элементы, центры которых расположены очень близко друг к другу и ширина которых приблизительно одинакова. Появление таких ситуаций во многом зависит от выбранного количества элементов и их начальных параметров. Причина ухудшения качества обучения в такой ситуации заключается в том, что в градиентном алгоритме предполагается, что на выходное значение РБНС в каждой точке рабочей области в основном влияет только один элемент. При наличии нескольких таких элементов изменение их параметров в соответствии с градиентным алгоритмом не всегда приводит к уменьшению ошибки обучения.

Для определения ситуаций, когда параметры некоторых элементов становятся близкими друг к другу, было введено понятие коэффициента взаимного пресечения элементов. Для вычисления этого коэффициента для некоторого элемента РБНС необходимо найти второй элемент, центр которого расположен ближе всего к центру рассматриваемого элемента. Значение коэффициента взаимного пересечения определяется как сумма выходной величины текущего элемента в центре второго элемента и выходной величины второго элемента в центре текущего элемента:

$$\rho_i = \exp\left(-\frac{\sum_{j=1}^{n}(c_{ij}-c_{dj})^2}{2\sigma_i^2}\right) + \exp\left(-\frac{\sum_{j=1}^{n}(c_{ij}-c_{dj})^2}{2\sigma_d^2}\right),$$

где i – номер элемента, для которого вычисляется значение коэффициента взаимного пересечения; d – номер элемента, центр которого расположен ближе всего к центру элемента с номером i. Номер элемента d определяется по формуле

$$d = \arg\min_k \sqrt{\sum_{j=1}^{n}(c_{ij}-c_{kj})^2}\ .$$

Значение коэффициента взаимного пересечения находится в интервале (0; 2]. Коэффициент принимает максимальное значение в том случае, когда центры рассматриваемых элементов совпадают. В ходе экспериментов по аппроксимации различных двумерных функций с помощью РБНС было определено, что ошибка РБНС начинает увеличиваться в том случае, когда максимальное значение коэффициента взаимного пересечения превышает 1,9. Поэтому для достижения максимального качества обучения РБНС необходимо ограничить увеличение значения коэффициента взаимного пересечения выше 1,9.

С целью исключения недостатков классического градиентного алгоритма обучения РБНС был разработан модифицированный градиентный алгоритм, блок-схема которого показана на рисунке 5. Блоки, которые отсутствуют в классическом алгоритме, отмечены на рисунке звездочками. Основные отличия от классического алгоритма заключаются в следующем:

1. Добавлены правила изменения структуры РБНС в процессе обучения (блок 2). В начале обучения РБНС не содержит элементов. По мере необходимости новые элементы добавляются, а неиспользуемые элементы удаляются.

2. Уменьшены вычислительные затраты, требуемые для каждого цикла обучения. Это достигается благодаря тому, что изменение параметров осуществляется не для всех элементов, как в классическом алгоритме, а только для элементов, выходная величина которых в рассматриваемой точке больше величины $\theta_{изм}$ (блоки 4 и 5).

3. Исключена возможность возникновения ситуации, когда параметры некоторых элементов практически совпадают. Для этого вычисленные величины Δc_{ij} и $\Delta\sigma_i$ уменьшаются в том случае, если коэффициент взаимного пересечения элементов превышает пороговую величину $\rho_{гр}$, равную 1,9 (блоки 7, 8, 12, 13).

Изменение структуры РБНС за счет добавления или удаления элементов приводит к изменению выходного значения РБНС только в окрестности центра добавляемого или удаляемого элемента, а не во всей рабочей области, как в случае с изменением структуры многослойного перцептрона. Поэтому добавление и удаление элементов РБНС возможно осуществлять в процессе обучения без необходимости запуска процесса обучения с самого начала.

Рассмотрим пример аппроксимации двумерной функции

$$f(x_1, x_2) = \sin(\frac{x_1^2}{2} - \frac{x_2^2}{4} + 3) \cdot \cos(2x_1 + 1 - \exp(-x_2))$$

на участке $x_1 \in [-1;1], x_2 \in [-1;1]$ с помощью РБНС. Поверхность данной функции показана на рисунке 6. При использовании классического градиентного алгоритма перед началом обучения была задана структура РБНС в виде 36 элементов с начальной шириной σ_0=0,2, равномерно распределенных в рабочей области. После приблизительно одного миллиона циклов обучения среднеквадратическая ошибка обучения перестала уменьшаться и достигла значения $1{,}554 \cdot 10^{-3}$.

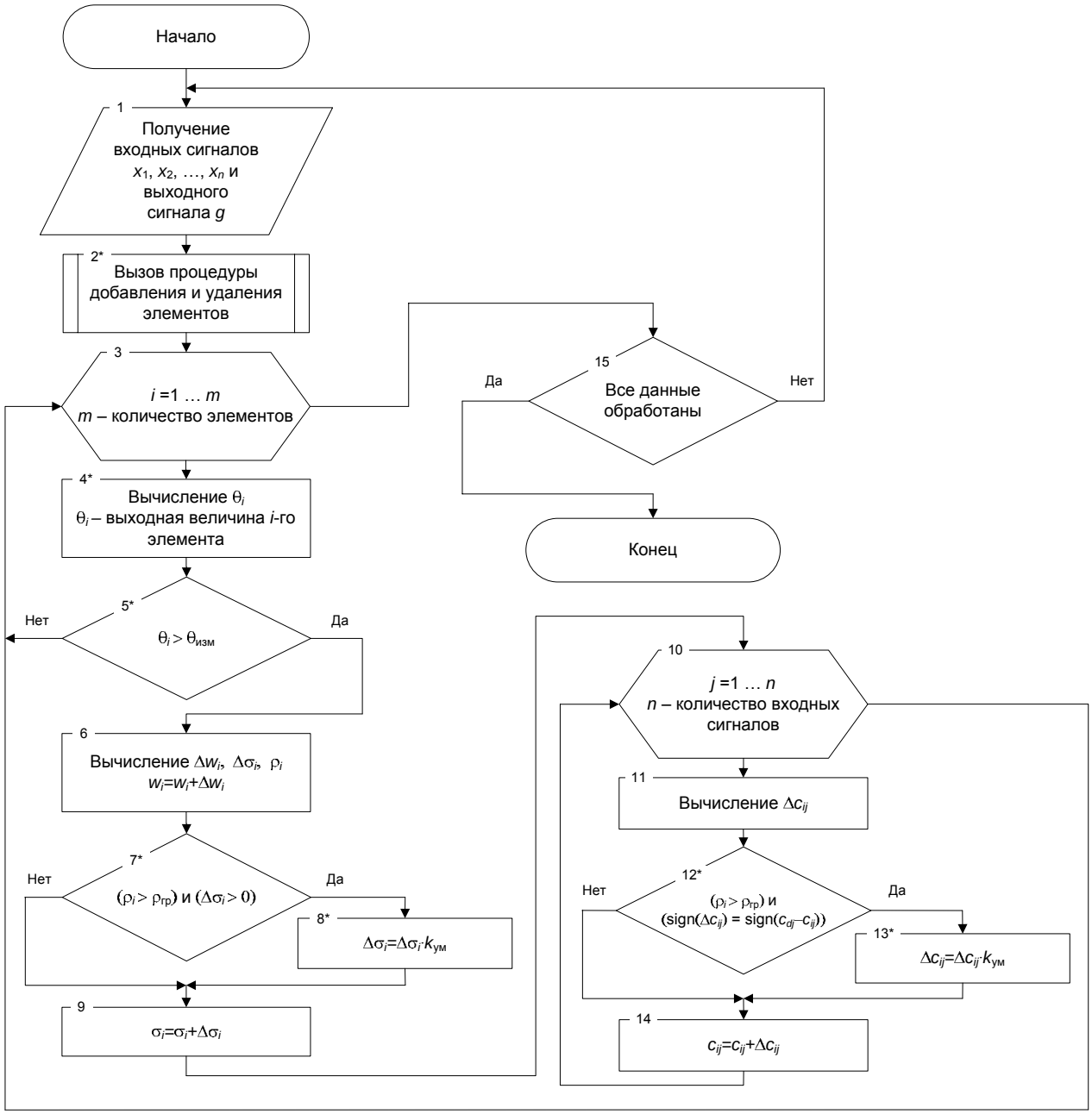

Рисунок 5 – Блок-схема модифицированного градиентного алгоритма обучения РБНС

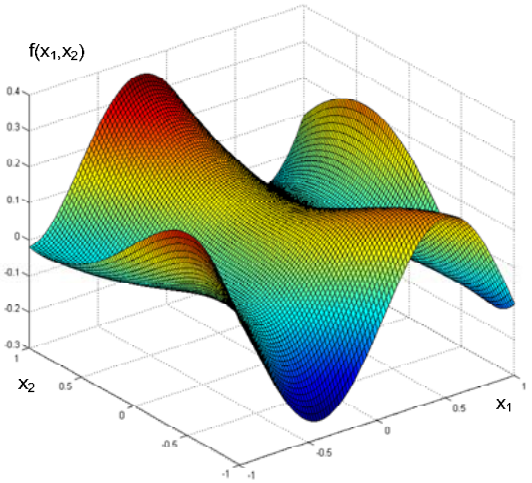

Рисунок 6 – Поверхность функции $f(x_1,x_2)$

При использовании модифицированного градиентного алгоритма структура РБНС была определена автоматически в процессе обучения. После приблизительно трех миллионов циклов обучения количество элементов увеличилось до 30, а среднеквадратическая ошибка обучения составила $1{,}225 \cdot 10^{-3}$. Результаты обучения РБНС показаны на рисунке 7. Таким образом, можно сделать вывод, что даже при меньшем количестве элементов модифицированный градиентный алгоритм позволяет достичь меньшей ошибки обучения по сравнению с классическим градиентным алгоритмом за счет динамического формирования структуры нейронной сети, но при этом требуется большее количество вычислительных ресурсов.

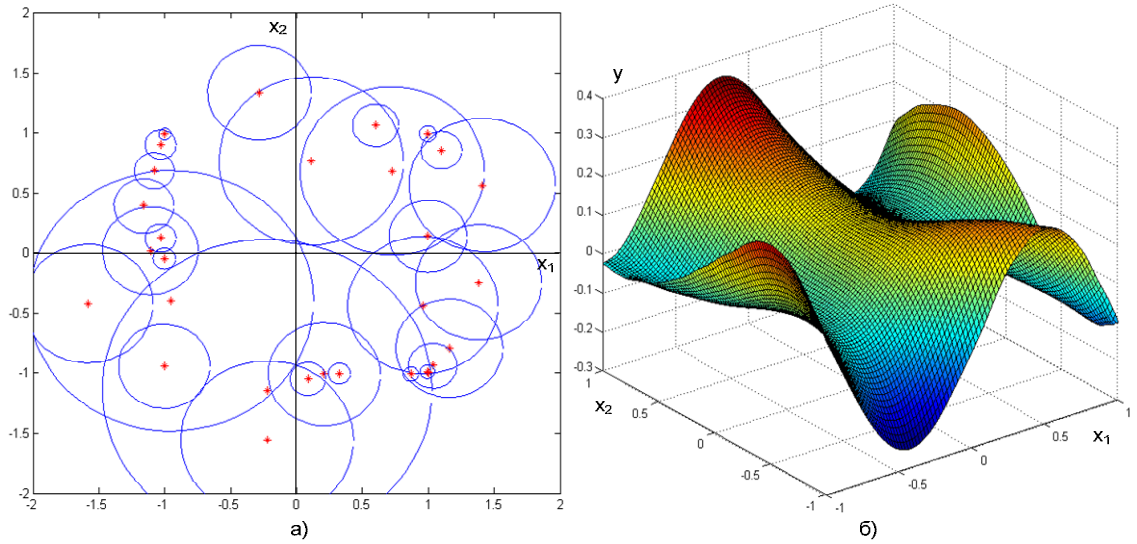

Рисунок 7 – Результат аппроксимации функции $f(x_1,x_2)$:
а) распределение элементов; б) поверхность, показывающая зависимость выхода РБНС от входных значений

Разработанный модифицированный градиентный алгоритм обучения РБНС позволяет заменить в обобщенной структурной схеме RL-САУ матрицу вещественных чисел, используемую в блоке УУО для представления функции оценки воздействия, на РБНС. Разработанная обобщенная структурная схема нейросетевой RL-САУ показана на рисунке 8. Импульсный элемент ИЭ, анализирующее устройство и экстраполятор работают по тем же алгоритмам, как и в структурной схеме RL-САУ, представленной во второй главе. Основные отличия данной схемы от структурной схемы RL-САУ заключаются в следующих блоках:

1. Квантователь заменен на блок нормализации (БН), формирующий вектор состояния внешней среды $\overline{S}[k]$, каждый элемент которого масштабирован к интервалу [-1;1].

2. УУО и УУА работают с функцией оценки воздействия, представленной не матрицей вещественных чисел, а с помощью РБНС.

3. Выходной сигнал УУО $a[k]$ представляет собою не сигнал управления, а изменение сигнала управления $u[k]$.

Значение управляющего сигнала $u[k]$ рассчитывается как сумма управляющего сигнала на предыдущем такте $u[k\text{-}1]$ и изменения этого сигнала

$a[k]$. В таком случае для обеспечения свойства марковости вектора состояния внешней среды на вход блока нормализации подается значение управляющего сигнала на предыдущем такте $u[k\text{-}1]$. Такой способ формирования сигнала $u[k]$ обеспечивает возможность формирования сложных управляющий воздействий на ОУ при ограниченном количестве возможных значений сигнала $a[k]$. На рисунке 9 приведен пример формирования синусоидального воздействия $u[k]$ в интервале [-5; 5] при пяти возможных значениях сигнала $a[k]$: минус 0,3; минус 0,1; 0; 0,1 и 0,3.

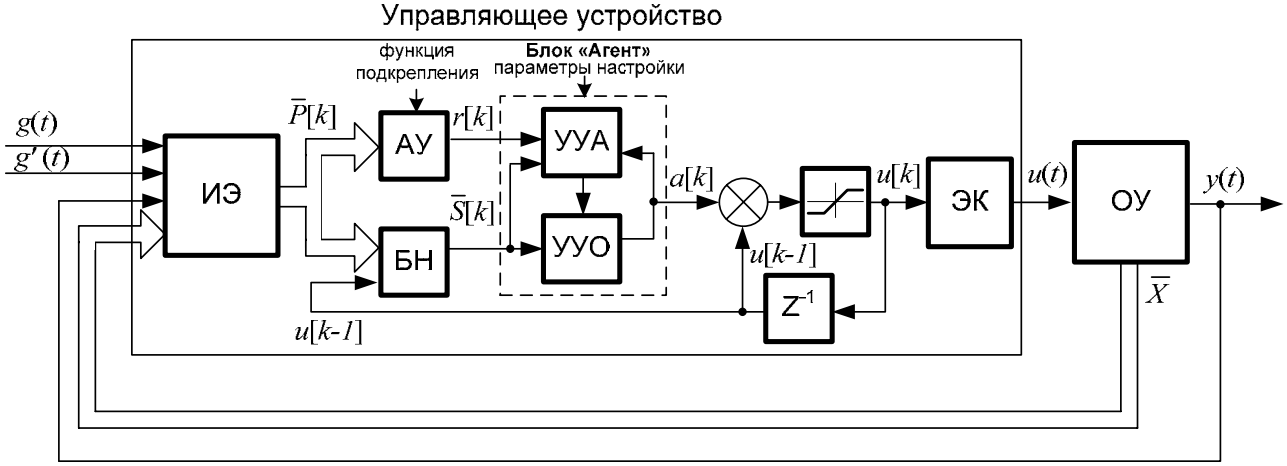

Рисунок 8 – Обобщенная структурная схема нейросетевой RL-САУ

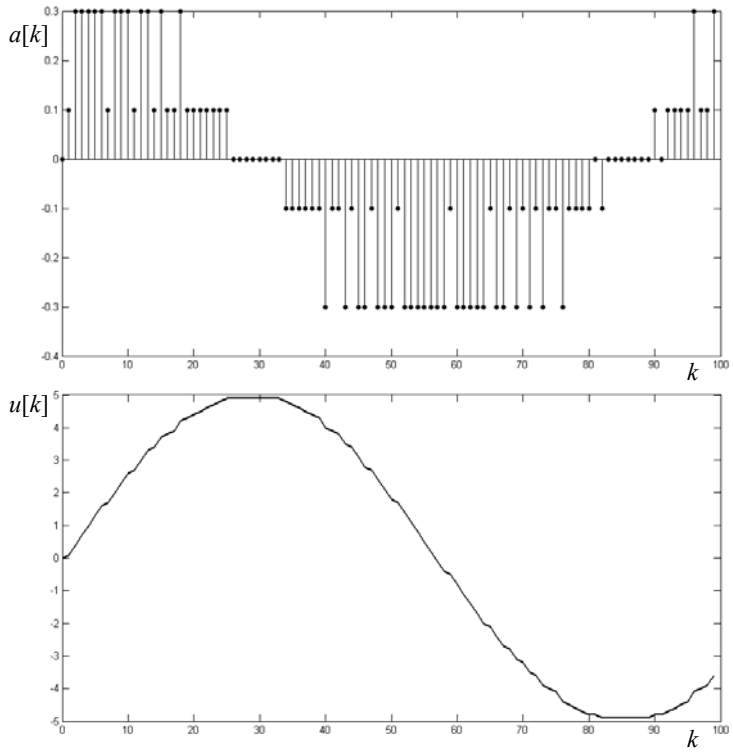

Рисунок 9 – Примеры графиков сигналов $a[k]$ и $u[k]$

В блоке «Агент» осуществляется определение функции оценки воздействия, которая представлена с помощью РБНС, обучаемой модифицированным градиентным алгоритмом. На вход РБНС подается N_s+1 сигналов (N_s – количество элементов вектора $\overline{S}[k]$): элементы вектора

состояния внешней среды $\overline{S}[k]$ и сигнал, соответствующий проверяемому выходному сигналу a. Последний сигнал, как и элементы вектора $\overline{S}[k]$, должен быть масштабирован к интервалу [-1;1]. Значение функции оценки воздействия определяется выражением

$$Q(a,\overline{S}[k]) = y(x_1, x_2, ..., x_n),\ x_i = s_i[k],\ i = \overline{1, n-1},\ x_n = 2\frac{a - a_{min}}{a_{max} - a_{min}} - 1,$$

где $y(x_1, x_2, ..., x_n)$ – выходное значение РБНС; x_i – входные сигналы РБНС; n – количество входных сигналов РБНС, которое определяется выражением $n=N_s+1$; a_{min} и a_{max} – минимальное и максимальное значения элементов множества возможных выходных сигналов A.

Для выбора выходного сигнала на текущем шаге $a[k]$ сначала определяется наилучшее значение $a^*[k]$, которое в соответствии с текущим значением функции оценки воздействия характеризуется наибольшим значением суммарной величины подкрепления. Для определения наилучшего значения выходного сигнала $a^*[k]$ рассматриваются все возможные выходные сигналы a из множества A и среди них выбирается то значение, которое соответствует максимальному значению оценки воздействия $Q(a, \overline{S}[k])$:

$$a^*[k] = \arg\max_{a \in A}(Q(a, \overline{S}[k])).$$

Для комбинирования процессов исследования внешней среды и использования накопленных знаний в блоке «Агент» используется «ε-жадная» стратегия управления. В соответствии с этой стратегией на каждом такте работы УУ определяется случайное значение величины e из диапазона [0,1). В зависимости от соотношения этой величины и параметра настройки УУ ε для формирования выходного сигнала выбирается либо наилучшее значение $a^*[k]$, либо случайно выбранное значение из всех возможных:

$$a[k] = \begin{cases} a^*[k], & \text{если } e \geq \varepsilon; \\ a_j,\ j = \lfloor rand \cdot N_A \rfloor + 1 & \text{иначе,} \end{cases}$$

где $rand$ – случайное число в диапазоне [0,1); N_A – количество элементов в множестве возможных выходных сигналов A; j – случайное целое число в диапазоне $[1,N_A]$, определяющее номер случайно выбранного выходного сигнала.

После того, как выходной сигнал $a[k]$ на текущем такте k сформирован, на следующем такте (k+1) УУА осуществляет коррекцию функции оценки воздействия в соответствии с алгоритмом Q-обучения и с учетом сигнала подкрепления $r[k+1]$ и вектора состояния внешней среды $\overline{S}[k+1]$. Для этого определяется наилучший сигнал $a^*[k+1]$ и вычисляется ошибка временной разности td:

$$td = r[k+1] + \gamma \cdot Q(a^*[k+1], \overline{S}[k+1]) - Q(a[k], \overline{S}[k]).$$

Ошибка временной разности используется для коррекции значения функции оценки воздействия в точке, определяемой аргументами $a[k]$ и $\overline{S}[k]$. Для этого вызывается процедура обучения РБНС по модифицированному

градиентному алгоритму в точке $(x_1,x_2,\ldots,x_n)$ новому значению РБНС y^*, которое определяется выражением

$$y^* = Q(a[k], \overline{S}[k]) + \alpha \cdot td,$$

где $\alpha \in [0,1]$ – параметр обучения.

Когда в процессе обучения ошибка временной разности будет приближаться к нулю, тогда наилучшие выходные сигналы $a^*[k]$ будут приводить к цели функционирования системы, то есть к максимизации суммарной величины подкрепления.

Для исследования нейросетевых RL-САУ было разработано программное средство «Исследование NRL-САУ», которое основано на исходном коде программного средства «Исследование RL-САУ». NRL является сокращением фразы Neuronet Reinforcement Learning. Изменения исходного кода произошли в модели УУ. Также изменился пользовательский интерфейс модуля «Управляющее устройство».

В программном средстве реализована возможность адаптивного изменения значения параметра алгоритма Q-обучения α в процессе обучения. Адаптивное значение параметра обучения α позволяет обеспечить устойчивость процесса обучения. Если средняя ошибка обучения РБНС большая, то параметр α постепенно уменьшается до минимального значения, и наоборот, если ошибка РБНС достаточно маленькая, то параметр α увеличивается до начального значения. Адаптивное значение параметра α определяется следующим выражением:

$$\alpha^{(k+1)} = \begin{cases} \alpha^{(k)}(1-\gamma_\alpha), & \text{если } (\varepsilon_{ср} > \varepsilon_2) \text{и} (\alpha^{(k)} > \alpha_{min}); \\ \alpha^{(k)}(1+\gamma_\alpha), & \text{если } (\varepsilon_{ср} < \varepsilon_1) \text{и} (\alpha^{(k)} < \alpha_0); \\ \alpha^{(k)} & \text{иначе,} \end{cases}$$

где $\alpha^{(k)}$ – значение параметра α на k-ом такте; α_{min} – минимальное значение параметра α; α_0 – начальное значение параметра α; γ_α – коэффициент изменения параметра α; ε_1 и ε_2 – нижняя и верхняя границы интервала допустимых значений средней ошибки $\varepsilon_{ср}$.

В программном средстве «Исследование NRL-САУ» были проведены экспериментальные исследования систем управления линейными и нелинейными ОУ второго, третьего и четвертого порядков, которые показали, что нейросетевая RL-САУ способна адаптироваться к неизвестным или изменяющимся свойствам ОУ для достижения цели функционирования. Рассмотрим пример управления ОУ «Акробот», который представляет собою два звена, соединенные между собою шарниром (рисунок 10). Первое звено соединено свободным концом шарниром с неподвижной точкой. Управляющим воздействием на данный ОУ является момент вращения M, приложенный ко второму звену. Выходной величиной ОУ является угол отклонения первого звена от вертикали θ_1.

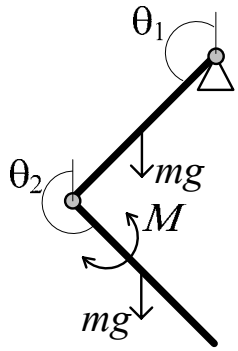

Рисунок 10 – ОУ «Акробот»

Математическая модель ОУ представлена системой нелинейных дифференциальных уравнений

$$\begin{cases} \dfrac{dx_1}{dt} = x_3; \\ \dfrac{dx_2}{dt} = x_4; \\ \dfrac{dx_3}{dt} = 0{,}19(2x_4^2\sin(x_2 - x_1) + 29{,}4\sin x_1 - 0{,}2x_3 - u) - \\ \qquad -2\cos(x_1 - x_2)(2x_3^2\sin(x_1 - x_2) + 9{,}8\sin x_2 - 0{,}2x_4 + u) - \\ \qquad -\dfrac{4(\cos(x_1 - x_2))^2(2x_4^2\sin(x_2 - x_1) + 29{,}4\sin x_1 - 0{,}2x_3 - u)}{5{,}33(7{,}11 - 4(\cos(x_1 - x_2))^2)}; \\ \dfrac{dx_4}{dt} = 5{,}33(2x_3^2\sin(x_1 - x_2) + 9{,}8\sin x_2 - 0{,}2x_4 + u) - \\ \qquad -\dfrac{2\cos(x_1 - x_2)(2x_4^2\sin(x_2 - x_1) + 29{,}4\sin x_1 - 0{,}2x_3 - u)}{7{,}11 - 4(\cos(x_1 - x_2))^2}; \\ y = x_1. \end{cases}$$

Целью управления является достижение нулевого значения выходной величины, что соответствует верхнему вертикальному положению первого звена. Максимальное и минимальное значения управляющего воздействия *u* равны 2 и минус 2 соответственно. Возможные изменения управляющего воздействия равны минус 2, 0 и 2. Таким образом, управляющее воздействие не может сразу измениться с максимального до минимального значения. Управляющее воздействие *u*=2 достаточно мало, чтобы перевести первое звено в требуемое положение, и приводит лишь к отклонению звеньев на небольшой угол. Поэтому единственный способ достижения требуемого состояния заключается в постепенном раскачивании звеньев.

За 5815 мин. модельного времени, что соответствует одному часу реального времени при моделировании на компьютере среднего класса, система научилась формировать последовательность управляющих воздействий продолжительностью 34 с, которая переводила ОУ в требуемое положение. На рисунке 11 показан график управляющего воздействия и положение звеньев через каждые 3 с. Серым цветом показаны промежуточные положения звеньев.

После обучения РБНС содержала 223 элемента, для хранения которых необходимо всего лишь около 20 килобайт оперативной памяти. Таким образом, результаты данного эксперимента подтверждают, что для нейросетевой RL-САУ отсутствует проблема экспоненциального роста объема требуемой памяти и времени обучения с ростом порядка ОУ.

В результате исследований нейросетевых RL-САУ с различными линейными и нелинейными ОУ были сделаны следующие выводы:

1. Нейросетевая RL-САУ способна формировать управляющие воздействия в соответствии с выбранным критерием функционирования системы при неизвестной или меняющейся математической модели ОУ.

2. В нейросетевой RL-САУ существует возможность исключить резкие переключения управляющего воздействия за счет определения возможных значений его изменения и границ изменения. Таким образом, при ограниченном количестве выходных значений УУО способно формировать плавно изменяющееся управляющее воздействие, которое может быть реализовано на реальных исполнительных механизмах.

3. Использование РБНС для представления функции оценки воздействия в нейросетевой RL-САУ позволяет устранить проблему, связанную с экспоненциальным ростом объема требуемой памяти при увеличении количества элементов во входном векторе. Например, для ОУ второго порядка потребовалось около 10 килобайт оперативной памяти, а для нелинейного ОУ четвертого порядка – около 20 килобайт. При использовании модифицированного градиентного алгоритма обучения РБНС структура нейронной сети определяется в процессе обучения.

4. Для нейросетевой RL-САУ актуальной является проблема большого времени обучения и переобучения УУ при изменении параметров ОУ, поэтому на данный момент практическое применение RL-САУ может заключаться в определении последовательности управляющих воздействий для перевода ОУ из исходного состояния в требуемое при известной математической модели ОУ.

Основные результаты работы

В результате выполнения диссертационной работы получены следующие основные научные и практические результаты и сделаны следующие выводы.

1. Разработан модифицированный градиентный алгоритм обучения РБНС, в котором введено понятие коэффициента взаимного пересечения элементов. Основная особенность алгоритма заключаются в динамическом определении структуры РБНС в процессе обучения.

2. Разработана обобщенная структурная схема нейросетевой RL-САУ, функционирующей на основе метода подкрепляемого обучения с применением РБНС для представления функции оценки воздействия. Определены алгоритмы работы структурных блоков.

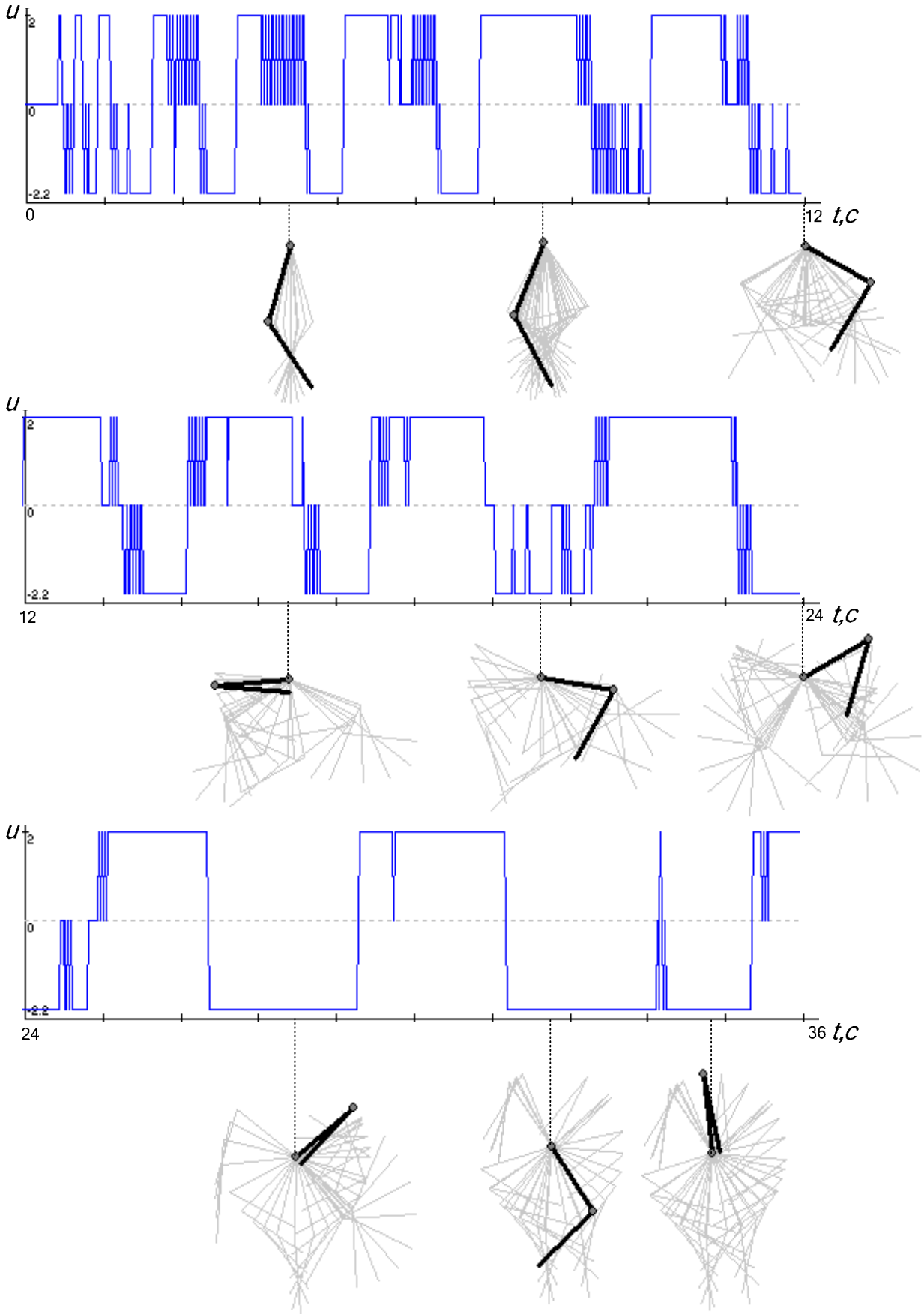

Рисунок 11 – График управляющего воздействия на ОУ «Акробот», переводящего ОУ в требуемое состояние

3. Разработано программное средство «Исследование NRL-САУ», предназначенное для моделирования и исследования нейросетевых RL-САУ. Программное средство позволяет задавать математическую модель ОУ в виде системы дифференциальных уравнений, определять вид и параметры задающего воздействия, задавать параметры настройки УУ, управлять процессом моделирования, отображать на экране значения всех моделируемых сигналов и их графики, определять значения показателей качества управления, сохранять результаты моделирования в файлы.

4. Предложены рекомендации по настройке параметров УУ в процессе функционирования RL-САУ. Предложен алгоритм адаптивного изменения значения параметра обучения, обеспечивающий устойчивость процесса обучения при использовании РБНС для представления функции оценки воздействия.

5. Результаты экспериментальных исследований нейросетевой RL-САУ с линейными и нелинейными объектами показали приемлемое качество управления и способность RL-САУ адаптироваться к неизвестным или изменяющимся параметрам ОУ.

Перечень публикаций по теме диссертации

1. **Вичугов, В.Н.** Нейросетевой метод подкрепляемого обучения в задачах автоматического управления // Известия Томского политехнического университета, 2006. – т.309, № 7. – С. 92-96.

2. **Вичугов, В.Н.** Метод подкрепляемого обучения в задачах автоматического управления / В.Н. Вичугов, Г.П. Цапко // Известия Таганрогского государственного радиотехнического университета, 2007. – № 3. – С. 171-174.

3. **Вичугов, В.Н.** Применение метода «Reinforcement Learning» в задачах автоматического управления / В.Н. Вичугов, С.Г. Цапко // Современные техника и технологии: Труды XI Международной научно-практической конференции студентов и молодых ученых. В 2 т. – Т. 2 – г. Томск, ТПУ, 28 марта – 1 апреля 2005 г. – Томск: Изд-во ТПУ, 2005. – С. 127-129.

4. **Вичугов, В.Н.** Представление Q-функций в RL-САУ на основе искусственной нейронной сети // Современные техника и технологии: Труды XII Международной научно-практической конференции студентов и молодых ученых – Томск, 27-31 марта 2006. – Томск: ТПУ, 2006. – С. 41-43.

5. **Вичугов, В.Н.** Моделирование адаптивных систем управления на основе подкрепляемого обучения / В.Н. Вичугов, Г.П. Цапко //Труды международных научно-технических конференций «Интеллектуальные системы» (IEEE AIS-06) и «Интеллектуальные САПР» (CAD-2006) – Дивноморское, 3-10 сентября 2006. – Москва: Физматлит, 2006. – С. 153-158.

6. **Вичугов, В.Н.** Применение метода подкрепляемого обучения для управления маятником // Высокие технологии, фундаментальные и прикладные исследования, образование: Сборник трудов Четвертой международной научно-практической конференции «Исследование, разработка и применение высоких

технологий в промышленности» - Санкт-Петербург, 2-5 октября 2007. – СПб: Изд-во Политехн. ун-та, 2007. – С. 309-311.
7. **Вичугов, В.Н.** Моделирование нейросетевых систем управления с использованием генетических алгоритмов обучения / В.Н. Вичугов, А.А. Вичугова // Имитационное моделирование. Теория и практика (ИММОД-2007): Сборник докладов третьей Всероссийской научно-практической конференции – Санкт-Петербург, 17-19 октября 2007. – СПб: ЦНИИТС, 2007. – С. 245-248.
8. **Вичугов, В.Н.** Применение генетических алгоритмов в нейросетевых системах автоматического управления / В.Н. Вичугов, А.А. Вичугова // Научная сессия МИФИ-2007. IX Всероссийская научно-техническая конференция «Нейроинформатика-2007» – Москва, 23-26 января 2007. – Москва: МИФИ, 2007. – С. 168-176.
9. **Вичугов, В.Н.** Адаптивные системы автоматического регулирования на основе нейронных сетей с применением генетических алгоритмов / В.Н. Вичугов, А.А. Вичугова // Современные техника и технологии: Труды XIII Международной научно-практической конференции студентов, аспирантов и молодых ученых – Томск, 26-30 марта 2007. – Томск: ТПУ, 2007. – С. 301-303.
10. **Вичугов, В.Н.** Нейросетевые системы управления с применением генетических алгоритмов / В.Н. Вичугов, А.А. Вичугова // Молодежь и современные информационные технологии: Сборник трудов V Всероссийской научно-практической конференции студентов, аспирантов и молодых ученых – Томск, 27 февр. – 1 марта 2007. – Томск: Изд. ТПУ, 2007. – С. 365-367.
11. **Вичугов, В.Н.** Алгоритм генетической настройки нейросети в задачах управления / В.Н. Вичугов, А.А. Вичугова // Молодежь и современные информационные технологии: Сборник трудов VI Всероссийской научно-практической конференции студентов, аспирантов и молодых ученых – Томск, 26-28 февраля 2008. – Томск: СПб Графикс, 2008. - с. 335-336.
12. **Вичугов, В.Н.** Поиск оптимальных параметров нейросети для решения задач управления / В.Н. Вичугов, А.А. Вичугова // Современные техника и технологии: Труды XIV Международной научно-практической конференции студентов, аспирантов и молодых ученых в 3-х томах – т. 3 – Томск, 24-28 марта 2008. – Томск: ТПУ: Изд. ТПУ, 2008. – С. 266-267.
13. **Vichugov, V.N.** Application of Reinforcement Learning in Control System Development / V.N. Vichugov, G.P. Tsapko, S.G. Tsapko // The 9-th Russian-Korean International Symposium on Science and Technology (KORUS-2005): Proceedings – Novosibirsk State Technical University, 26 June – 2 July 2005. – Novosibirsk: NSTU, 2005. – P. 732-733.
14. **Vichugov, V.N.** Neural-Based Reinforcement Learning in Control Systems // Мехатроника: устройства и управление: Материалы II российско-корейского научно-технического семинара – Томск, 18 марта 2008. – Томск: ТПУ, 2008. – с. 18-19.