

Министерство образования и науки Российской Федерации  
Федеральное государственное автономное образовательное учреждение  
высшего образования  
«Санкт-Петербургский политехнический университет Петра Великого»  
Институт компьютерных наук и технологий  
Кафедра «Системы и технологии управления»

Работа допущена к защите

Заведующий кафедрой СТУ

\_\_\_\_\_ В.П. Шкодырев

«\_\_» \_\_\_\_\_ 2016 г.

## **ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА МАГИСТЕРСКАЯ ДИССЕРТАЦИЯ**

### **СЛЕДЯЩАЯ СИСТЕМА УПРАВЛЕНИЯ С НЕЙРОСЕТЕВОЙ АДАПТАЦИЕЙ ПАРАМЕТРОВ ПИД-РЕГУЛЯТОРА**

направление 27.04.04 – Управление в технических системах

магистерская программа 27.04.04\_02 «Автоматизация технологических процессов и  
производств»

Выполнил

студент гр. 63503/2

<

>

М.А. Смирнов

Научный руководитель

к.т.н., доцент

<

>

Н.В. Ростов

Санкт-Петербург

2016

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ  
РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное автономное образовательное учреждение высшего  
образования

«Санкт-Петербургский политехнический университет Петра Великого»  
Институт компьютерных наук и технологий  
Кафедра Системы и технологии управления

УТВЕРЖДАЮ  
Зав. кафедрой СТУ д.т.н., профессор  
\_\_\_\_\_ В.П. Шкодырев

«\_\_» \_\_\_\_\_ 2016 г.

ЗАДАНИЕ

на выполнение магистерской диссертации

Институт компьютерных наук и технологий  
по направлению 27.04.04 – Управление в технических системах  
магистерская программа 27.04.04\_02 «Автоматизация технологических процессов и  
производств»

студента \_\_\_\_\_ Смирнова Михаила Александровича \_\_\_\_\_

1. Тема магистерской диссертации: \_\_\_\_\_ Следящая система управления с  
нейросетевой адаптацией параметров ПИД-регулятора \_\_\_\_\_

2. Срок сдачи студентом законченной работы:

«10» июня 2016 г

3. Исходные данные к магистерской диссертации: \_\_\_\_\_

\_\_\_\_\_ Литература по прилагаемому списку \_\_\_\_\_

\_\_\_\_\_ Значения параметров следящей системы \_\_\_\_\_

\_\_\_\_\_ Информационные материалы по нейронным сетям и нейроуправлению \_\_\_\_\_

4. Содержание расчётно-пояснительной записки (перечень подлежащих

разработке вопросов): \_\_\_\_\_

Введение

1) Обзор известных методов нейросетевого управления

2) Моделирование прототипной следящей системы с неадаптивным ПИД-регулятором положения

3) Разработка алгоритмов нейросетевой адаптации параметров ПИД-регулятора

4) Вопросы технической реализации

Заключение

Список литературы

5. Перечень графического материала (с точным указанием обязательных иллюстраций): \_\_\_\_\_

1) Цели и задачи. (1 плакат)

2) Обзор методов нейроуправления. (2 плаката)

3) Схема и динамические процессы прототипной следящей системы. (3 плаката)

4) Динамические процессы системы при нейросетевом управлении (6 плакатов).

5) Способы технической реализации. (2 плаката)

6) Заключение и выводы. (1 плакат)

6. Консультанты по магистерской диссертации:

7. Дата выдачи задания «15» декабря 2015 г.

Руководитель,  
к.т.н., доцент

\_\_\_\_\_

*подпись*

(\_\_\_\_\_)

*расшифровка*

Задание принял к исполнению « \_\_\_ » \_\_\_\_\_ 20\_\_ г.

Студент

\_\_\_\_\_

*подпись*

(\_\_\_\_\_)

*расшифровка*

## Реферат

УДК 62-503.54

### СЛЕДЯЩАЯ СИСТЕМА, НЕЙРОННЫЕ СЕТИ, АДАПТИВНЫЙ РЕГУЛЯТОР, НЕЙРОСЕТЕВОЕ УПРАВЛЕНИЕ

Диссертация, темой которой является «следящая система управления с нейросетевой адаптацией параметров пид-регулятора» содержит 64 страницы, 38 рисунков использовано 15 источников литературы.

Настоящая диссертация посвящена разработке алгоритма нейросетевого модуля адаптации параметров регулятора, а также алгоритма адаптивного нейросетевого регулятора.

Приводится обзор известных методов нейросетевого управления с рассмотрением достоинств и недостатков каждого из них.

Разработана модель следящей системы, ПИД-регулятор положения которой в одном случае получает значения коэффициентов усиления из модуля нейросетевого адаптации, а в другом случае заменен на нейросетевой адаптивный регулятор.

Рассмотрены возможные варианты программной и аппаратной реализации разработанных алгоритмов, приведены рекомендации по выбору способа реализации.

## **Abstract**

UDC 62-503.54

TRACKING SYSTEM, NEURAL NETWORK, ADAPTIVE CONTROLLER,  
NEURAL NETWORK CONTROL

This thesis, whose theme is "Tracking control system with PID controller parameters adaptation based on neural network" contains 64 pages, 38 drawings, used 15 literature sources.

The thesis is devoted to development of neural network algorithm module to adapt the controller parameters and algorithm of adaptive neural network controller.

A review of known methods of neural network control with consideration of the advantages and disadvantages of each.

A simulation of the tracking system, PID controller position which in one case gets gain values of the neural network module adaptation, and in another case, replaced by a neural network adaptive controller.

Possible options for hardware and software implementation of the developed algorithms are given recommendations on the choice of the method.

## Содержание

1. Обзор известных методов нейросетевого управления .....	10
1.1. Подражающее нейроуправление .....	11
1.2. Инверсное нейроуправление .....	12
1.3. Прогнозирующее модельное нейроуправление .....	15
1.4. Многомодульное нейроуправление на основе пар прямых и инверсных моделей .....	17
1.5. Гибридное нейроуправление .....	18
2. Моделирование прототипной следящей системы с неадаптивным ПИД-регулятором положения .....	21
2.1. Структура следящей системы .....	21
2.2. Математическое описание следящей системы .....	23
2.2.1. Модель двигателя постоянного тока .....	23
2.2.2. Силовой преобразователь .....	24
2.2.3. Регуляторы .....	25
2.2.4. Модели датчиков и редуктора .....	27
2.3. Настройка параметров регуляторов системы .....	28
2.4. Анализ динамических процессов системы .....	32
3. Разработка алгоритмов нейросетевой адаптации параметров ПИД-регулятора .....	36
3.1. Модификация следящей системы блоком нейросетевой адаптации .....	37
3.2. Математическое описание блока нейросетевой адаптации .....	38
3.3. Моделирование следящей системы с нейро-ПИД управлением .....	44
3.4. Замена ПИД-регулятора на нейросетевой регулятор .....	47
3.5. Математическое описание нейросетевого регулятора .....	48
3.6. Моделирование следящей системы с нейрорегулятором .....	52
4. Вопросы технической реализации .....	55
4.1. Инструменты программной реализации нейровычислений .....	55
4.2. Аппаратные средства реализации нейровычислений .....	57
Заключение .....	61
Список сокращений .....	62
Список литературы .....	63

## Введение

На сегодняшний день интеллектуальные системы управления (ИСУ) различными процессами и объектами являются перспективным направлением научных исследований и разработок. Однако, несмотря на то, что интеллектуальные системы изучаются и описываются ведущими учеными из разных стран, теоретические и особенно прикладные основы ИСУ еще далеки от завершения и имеют много пространства для расширения и развития. Среди прочих областей применения ИСУ можно выделить применение в системах управления и контроля сложными техническими объектами, включающих в себя элементы, основанные на применении нечеткой логики, искусственных нейронных сетей и экспертных систем. Такие системы функционируют при сложных условиях работы, таких как нестационарность параметров объекта управления, его нелинейность, существенные запаздывания, отсутствие полного контроля входных воздействий и состояний объекта, наличие неконтролируемых внешних воздействий и помех.

Современные машины, агрегаты, технологические линии и участки различных отраслей промышленности в большинстве случаев являются сложными объектами. Существенного повышения эффективности управления ими можно достигнуть путем применения адекватно сложных информационно-управляющих систем, какими и являются интеллектуальные системы.

Актуальность применения ИСУ обусловлена стремительным развитием технологий автоматизации, направленных на повышение эффективности контроля и управления производственными процессами, а также на улучшение качества готовой продукции предприятий.

Одним из передовых направлений развития ИСУ является построение систем, использующих нейросетевое управление (нейроуправление). Нейроуправление — это частный случай интеллектуального управления, при

котором в качестве инструмента для решения задач управления применяются искусственные нейронные сети (ИНС).

Интеллектуальные системы с нейроуправлением, находят все большее распространение в различных областях науки и производства. В качестве наиболее распространенных примеров применения можно привести следующие:

- диагностика и определение технического состояния промышленных установок;
- анализ качества выпускаемой продукции;
- управление технологическими процессами с целью оптимизации определенных критериев;
- управление плохо описанными объектами;
- управление объектами в условиях неопределенности или нестационарности параметров;
- распознавание образов в системах компьютерного зрения;
- распознавание голосовых команд пользователей;
- классификация объектов по различным критериям;
- многие другие задачи из различных сфер деятельности человека.

В данной работе объектом исследования является следящая система управления с ПИД-регулятором положения. Система с регулятором, настроенным под определенные параметры объекта управления, способна функционировать с соблюдением всех требований точности только в случае стационарности значений указанных параметров. Если же параметры объекта управления в процессе работы системы меняются, то показатели качества динамических процессов резка ухудшаются.

Для того чтобы система могла работать при переменных параметрах объекта управления, необходимо каким-либо образом варьировать параметры регулятора. При условии, что закон изменения параметров объекта заранее известен, то можно обеспечить точную автоматическую подстройку регулятора без потерь качества регулирования. Однако, в том



случае, если неизвестны законы изменения хотя бы некоторых из параметров, или их изменение носит случайный характер, то обеспечить автоматическую корректировку значений параметров регулятора на заранее вычисленные не представляется возможным. Для решения этой проблемы был выбран способ адаптации параметров ПИД-регулятора на основе искусственных нейронных сетей.

Целью диссертации является разработка алгоритма нейросетевой адаптации параметров ПИД-регулятора и моделирование следящей системы с изменяющимся во времени моментом инерции.

Задачи, решаемые в процессе выполнения диссертации:

- обзор методов нейруправления динамическими объектами;
- моделирование прототипной следящей системы с неадаптивным ПИД-регулятором положения;
- разработка алгоритма нейросетевой адаптации параметров ПИД-регулятора положения следящей системы;
- разработка алгоритма адаптивного нейросетевого регулятора;
- компьютерное исследование динамики следящей системы с использованием разработанных регуляторов;
- рассмотрение вопросов технической реализации адаптивных нейросетевых регуляторов с разработанными алгоритмами.

## 1. Обзор известных методов нейросетевого управления

Одной из областей применения искусственных нейронных сетей являются системы автоматического управления. Разработано уже немало методов управления с применением ИНС. Такие методы называются методами нейруправления. Сводная таблица методов нейруправления приведена ниже.

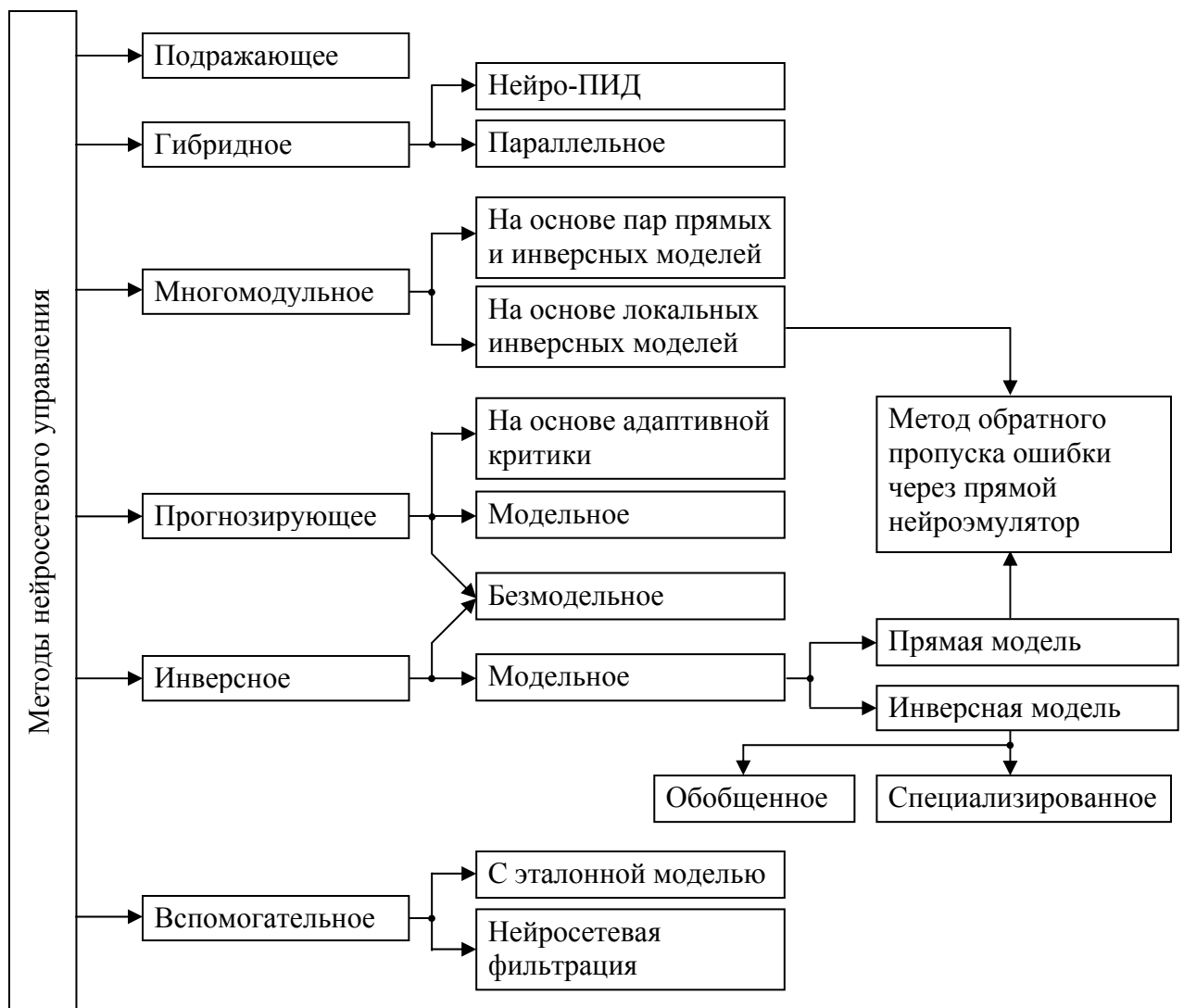


Рис. 1 Классификация методов нейруправления

Далее в кратком обзоре рассмотрим наиболее распространенные методы нейросетевого управления.

## 1.1. Подражающее нейруправление

Основным компонентом подражающего нейруправления является нейронная сеть, обученная по принципу супервизорного обучения с использованием существующего регулятора (например ПИД-регулятора). Иными словами, обучающей выборкой являются примеры динамики эталонного регулятора (или в редких случаях запись поведения человека-оператора). Обучающая выборка составляется из значений входов и выходов регулятора в процессе штатного управления объектом. Схема подражающего нейруправления представлена на рисунке 2: слева схема во время процесса обучения НС, справа – во время работы в качестве регулятора.

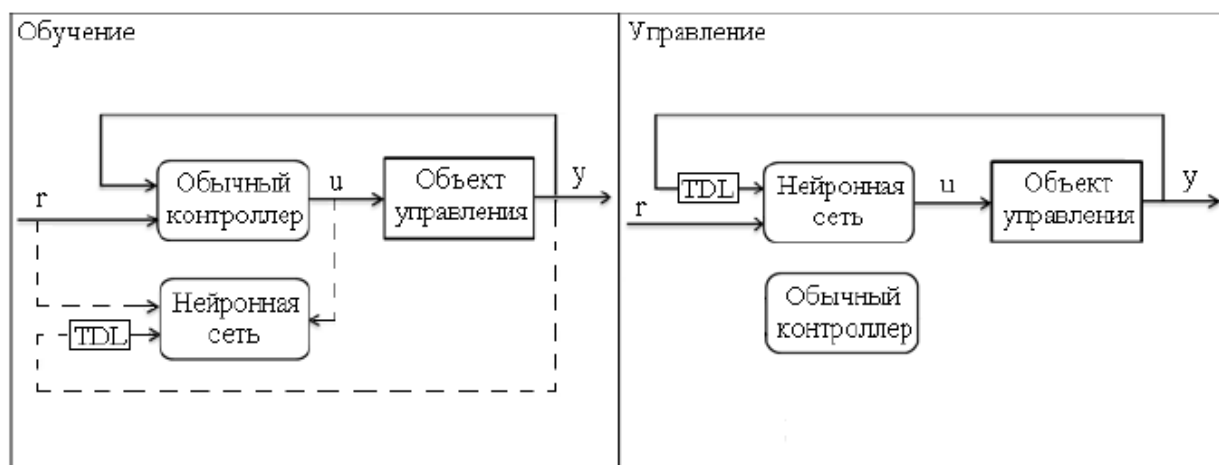


Рис. 2 Схема подражающего нейруправления

Нейронная сеть обучается по одному из методов (по методу обратного распространения ошибки, например), используя полученную выборку. После этого НС полностью воспроизводит динамику регулятора, на основе которой стоилась обучающая выборка, и может быть включена в контур управления вместо прежнего регулятора.

Подражающее нейруправления – самый простой способ применения нейронных сетей в САУ. Однако этот метод имеет ряд недостатков. Одним

из главных недостатков является необходимость наличия регулятора, заранее настроенного для управления имеющимся объектом, для построения выборки. К тому же, полученный таким образом нейросетевой регулятор не сможет обеспечить качество управления лучшее, чем исходный регулятор. В связи с вышесказанным, данный метод нейроуправления используют чаще всего для первичного обучения сети, с целью последующего дообучения по другим методикам.

## 1.2. Инверсное нейроуправление

Существует несколько вариантов рассматриваемого метода управления. Основным является обобщенное (или прямое) инверсное нейроуправление. Этот метод подразумевает обучение НС на основе данных, полученных с объекта управления.

В качестве входного сигнала ОУ и выходного значения обучающей выборки для НС используется некоторый случайный процесс. В качестве входных данных обучающей выборки нейронной сети используют выход ОУ. Для обучения используют метод обратного распространения ошибки.

После успешного обучения НС должна воспроизводить значения требуемого управляющего воздействия  $u$  в зависимости от желаемого выхода объекта  $y$ . Другими словами, НС должна вырабатывать управляющее воздействие, которое приведет объект управления в состояние, заданное уставкой.

Обученная таким образом НС называется инверсным нейроэмулятором. Схема обобщенного инверсного нейроуправления представлена на рисунке 3: слева схема в режиме обучения инверсного нейроэмулятора, справа – в режиме управления объектом.

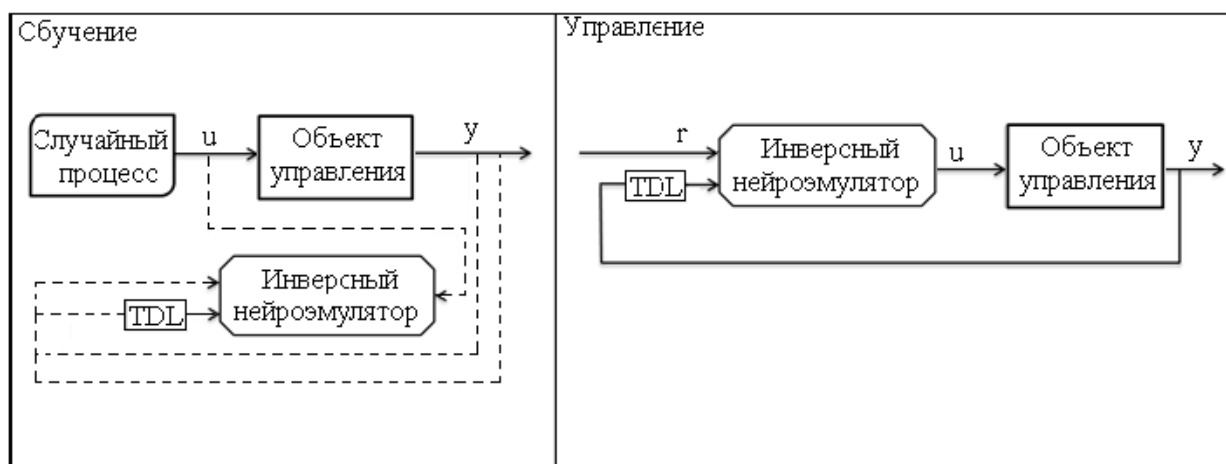


Рис. 3 Схема обобщенного инверсного нейроуправления

Как видно из рисунка 3, инверсный нейромодулятор включается в контур управления объектом в качестве регулятора.

Недостатком метода обобщенного инверсного нейроуправления является трудность в составлении обучающей выборки, выраженная в тщательном подборе идентифицирующего процесса. Также, трудности в обучении возникают, если инверсия ОУ является неоднозначной функцией. Это приводит процесс обучения в тупик из-за противоречий в обучающей выборке.

Достоинством метода является отсутствие необходимости в математической модели объекта.

Другой, более сложной разновидностью инверсного нейроуправления является метод обратного пропуска ошибки через прямой нейромодулятор, также известный в зарубежных источниках как Backpropagation Through Time (ВРТТ). Данный метод основан на применении сразу двух нейронных сетей, одна из которых выступает в роли регулятора, а другая - в роли прямого нейромодулятора.

Прямой нейромодулятор обучается таким образом, чтобы повторять поведение объекта управления, в зависимости от поступающих на его вход воздействий. Входными воздействиями для нейронной сети, содержащимися в обучающей выборке, являются значения случайного идентифицирующего

процесса и предыдущее состояние ОУ, а желаемые выходные значения – это выход ОУ при заданном входе. В итоге, сеть считается обученной, если при одинаковых входных значениях, поступивших на нее и на объект управления, их выходы будут отличаться не более чем на заданную малую величину.

После того, как прямой нейроэмулятор обучен и способен повторять динамику объекта, происходит обучение НС, выполняющей функции регулятора.

Обучение регулятора происходит в режиме он-лайн. На вход нейрорегулятора на  $k$ -ом такте поступает желаемое положение ОУ на следующем такте  $r(k+1)$ , при этом на выходе регулятора формируется управляющее воздействие  $u(k)$ . После отработки воздействия  $u(k)$ , ОУ переходит в состояние  $y(k+1)$ , а нейроэмулятор выдает значение  $\hat{y}(k+1)$ . На основании этих сигналов, вычисляется ошибка регулирования  $e = y - \hat{y}$ , которая по методу обратного распространения проходит через нейроэмулятор и нейрорегулятор. При этом, корректируются веса связей НС-регулятора, а веса нейроэмулятора не меняются, он лишь учитывается в качестве дополнительных слоев НС-регулятора.

Этот метод относится к инверсному нейроуправлению, т.к. механизм обратного распространения ошибки через прямой нейроэмулятор реализует локальную инверсную модель в текущей точке пространства состояний объекта управления.[3]

Схема метода обратного пропуска ошибки через прямой нейроэмулятор представлена на рисунке 4: слева показана схема в процессе обучения прямого нейроэмулятора, справа – схема обучения нейроконтроллера.

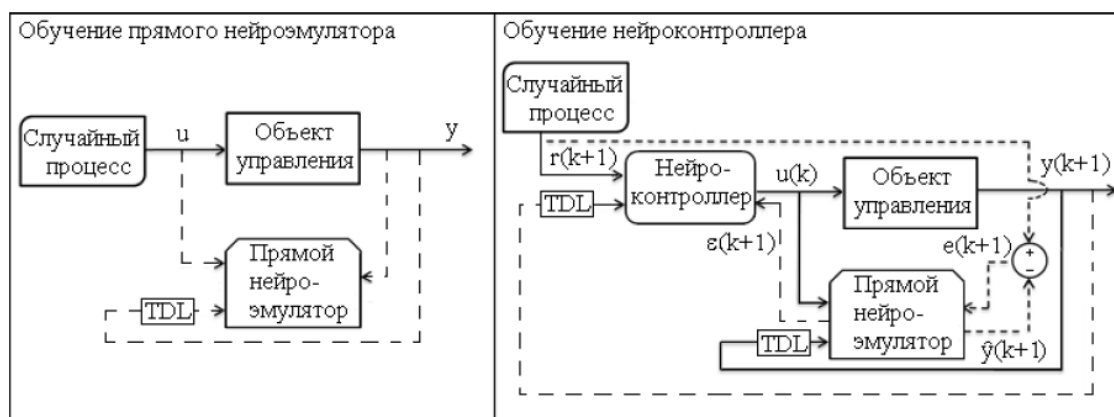


Рис. 4 Схема метода обратного пропуска ошибки через прямой нейроэмулятор

В качестве недостатка данного метода построения регулятора можно упомянуть необходимость создания и обучения сразу двух нейронных сетей. Достоинством можно назвать лучшее качество управления, по сравнению с методами, рассмотренными ранее.

### 1.3. Прогнозирующее модельное нейроуправление

Методы обучения нейросетевых регуляторов, при которых минимизируется отклонение текущего выхода ОУ от входного воздействия, не всегда хорошо справляются в задаче регулирования. Во-первых, если переходный процесс длится дольше, чем один такт, то НС, стремясь минимизировать рассогласование, может выдать большое управляющее воздействие, что приведет к перерегулированию. Во-вторых, в таких системах всегда существует запаздывание из-за обратной связи, что ухудшает качество регулирования.

В связи с этим, был разработан метод прогнозирующего модельного нейроуправления, который минимизирует функционал стоимости интегральной ошибки, прогнозируемой на несколько тактов вперед. Схема метода показана на рисунке 5.

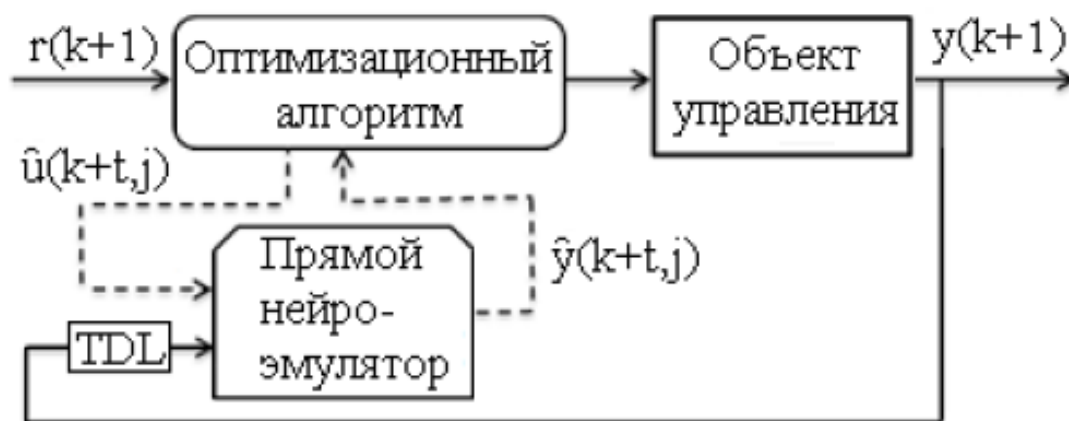


Рис. 5 Схема прогнозирующего модельного нейроуправления

Для прогнозирования будущего поведения ОУ и вычисления ошибок управления, используется прямой нейроэмулятор, обученный аналогично случаю обратного распространению ошибки через прямой нейроэмулятор (левая часть рисунка 4). Но отличие состоит в отсутствии обучаемого нейроконтроллера. Вместо него используется оптимизационный модуль, работающий в режиме реального времени.

На  $k$ -м такте оптимизационный модуль получает желаемую траекторию на несколько тактов вперед. После этого, он подает на прямой нейроэмулятор различные управляющие воздействия и прогнозирует поведение ОУ и вычисляет наилучшую стратегию управления. В результате выбирается оптимальное управляющее воздействие, приводящее ОУ к нужному положению на целевой траектории.

Минусом данного метода является невозможность его применения в системах с высокой частотой дискретизации, т.к. оптимизационный алгоритм за один такт не будет успевать находить оптимальное управляющее воздействие.



#### **1.4. Многомодульное нейруправление на основе пар прямых и инверсных моделей**

Многомодульные нейросистемы построены по принципу комитетов экспертов [3], т.е. состоят из отдельных модулей, каждый из которых решает свою отдельную подзадачу общей задачи. Итоговое управляющее воздействие складывается из совокупности управляющих значений отдельных модулей.

Рассматриваемый метод предполагает формирование поведения системы в процессе обучения, и корректировку поведения нейронных модулей на каждом такте нейруправления. В связи с этим, каждый нейросетевой модуль включает в себя два нейроэмулятора: прямой и обратный. Прямой нейроэмулятор проходит процесс обучения, идентичный рассмотренному в пункте 1.2.2., по схеме, изображенной на рисунке 4 слева. Инверсный нейроэмулятор обучается по схеме, представленной на рисунке 3 слева.

Схема многомодульного управления на основе пар прямых и инверсных моделей представлена на рисунке 6.

Функционирование системы на каждом такте состоит из двух этапов. Первым этапом работы является переоценка коэффициентов ответственности модулей, схема которого представлена на рисунке 6 слева. На входы прямых нейроэмуляторов каждого модуля поступает значения управляющего воздействия  $u(k-1)$  и состояния ОУ  $y(k-1)$  на предыдущем такте. По этим данным, каждый модуль производит свою оценку текущего положения ОУ, на основе которой вычисляется ошибка предвидения, по которой впоследствии вычисляется коэффициент предвидения (ответственности). На втором этапе происходит коллективное управление объектом, с использованием вычисленных коэффициентов ответственности. Эту задачу выполняют инверсные нейроэмуляторы, работающие в режиме обобщенного инверсного нейруправления. Результирующим воздействием на ОУ является

взвешенная сумма управляющих воздействий всех модулей системы, вклад каждого из которых пропорционален коэффициенту предвидения.

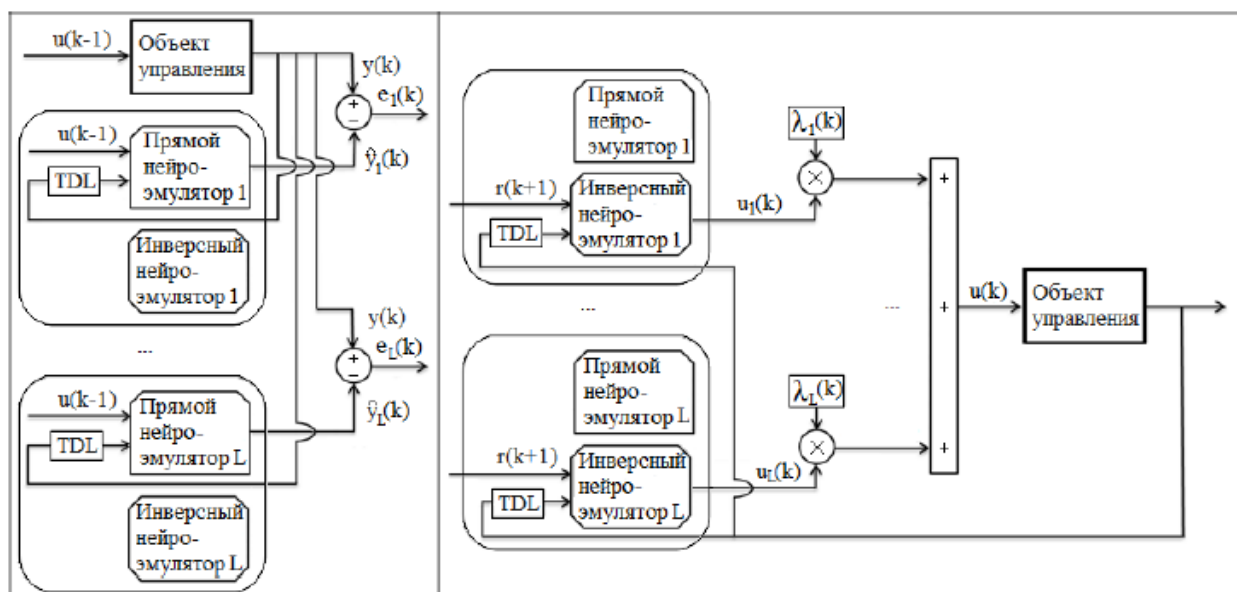


Рис. 6 Схема многомодульного управления на основе пар прямых и инверсных моделей

Подразумевается, что каждый модуль обучается на своем, существенно отличающемся от других, примере траектории состояний ОУ и специализируется именно на нем. В связи с этим, существенным недостатком многомодульного нейроуправления является трудность в составлении обучающих выборок для каждого нейроэмулятора всех модулей.

## 1.5. Гибридное нейроуправление

Гибридным нейроуправлением называют совместное функционирование нейросетевых регуляторов и стандартных регуляторов классической теории управления.

Одним из видов гибридного нейроуправления является параллельное гибридное нейроуправление, схема которого представлена на рисунке 7.

Существует несколько способов обучения и подключения нейросетевого регулятора к системе управления. Например, сначала контур управления объектом замыкается классическим регулятором. Далее, нейроконтроллер учится управлять этой замкнутой системой, после чего он подключается к системе и управляющие воздействия нейросетевого и обычного регуляторов складываются.

Либо в другом порядке: сначала контур управления замыкается нейросетевым регулятором, который проходит обучение. После этого настраивается обычный регулятор для управление этой системой. Настроенный регулятор подключается к системе и управляющие воздействия обоих регуляторов складываются.

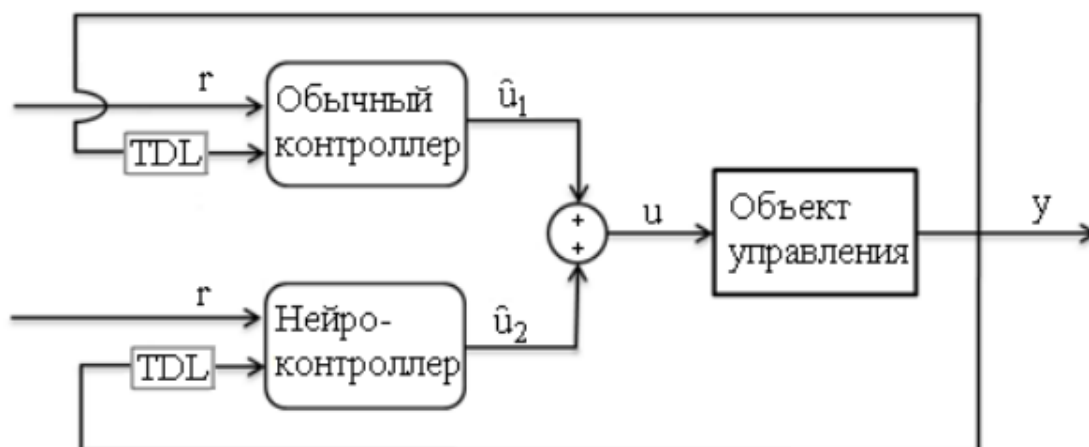


Рис. 7 Схема параллельного гибридного нейроуправления

Третьим вариантом параллельного гибридного нейроуправления является разграничение областей действия обычного регулятора и нейросетевого. В результате, на ОУ поступает управляющее воздействие либо от одного регулятора, либо от другого, в зависимости от положения объекта управления.

Параллельное гибридное нейроуправление является самым простым решением для перехода от обычных регуляторов к нейросетевым.

Другим методом гибридного нейроуправления является гибридное нейро-ПИД управление (или нейроморфное управление), сутью которого является самонастройка ПИД-регулятора с помощью искусственных нейронных сетей в режиме он-лайн.

Схема данного метода представлена на рисунке 8. На входе нейронная сеть получает уставку требуемого положения объекта и предыдущего его состояния. На выходе получают коэффициенты управления ПИД-регулятора  $K_p$ ,  $K_i$ ,  $K_d$ .

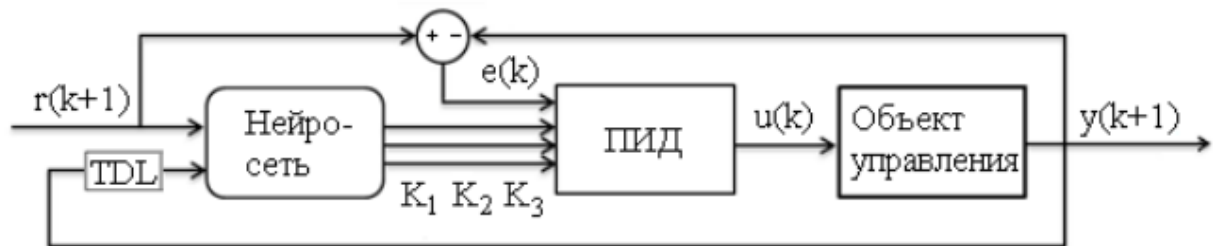


Рис. 8 Схема гибридного нейро-ПИД управления

Сгенерированные значения параметров вместе с текущей ошибкой управления поступают на ПИД-регулятор, который в свою очередь генерирует управляющее воздействие на объект управления.

Достоинством этого метода является то, что обычный ПИД-регулятор превращается в нелинейный, что обеспечивает более высокое качество управления нелинейными объектами.

Подводя итог вышесказанному, можно отметить что нейроуправление интенсивно развивается, разрабатывая все новые методы и структуры управления объектами.

В данной работе основным исследуемым методом будет являться последний из рассмотренных, т.е. метод гибридного нейро-ПИД управления.

## 2. Моделирование прототипной следящей системы с неадаптивным ПИД-регулятором положения

В качестве исследуемой системы принята следящая система управления с ПИД-регулятором положения и двигателем постоянного тока в качестве регулируемого объекта.

Для оценки динамических процессов системы с обычным ПИД-регулятором была построена модель средствами Simulink.

### 2.1. Структура следящей системы

Функциональная схема моделируемой следящей системы представлена на рис. 9.

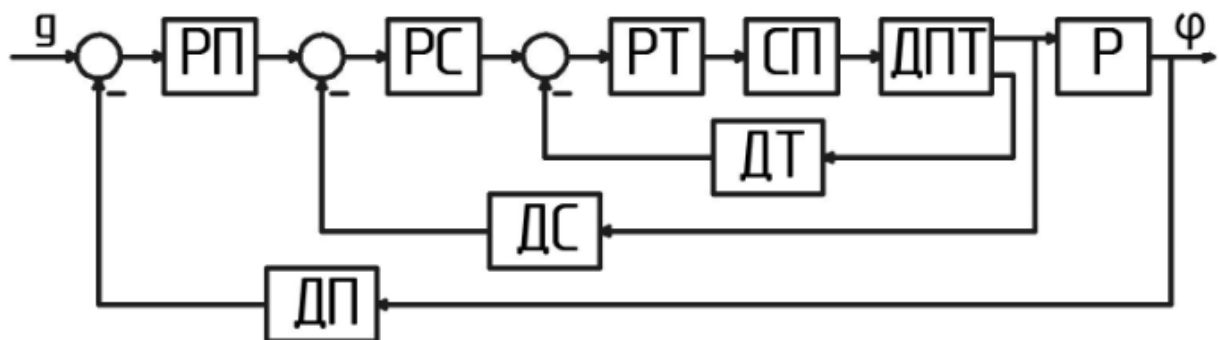


Рис. 9 Функциональная схема следящей системы

Система состоит из трех контуров управления двигателем постоянного тока: контура тока, контура скорости и контура положения. Соответственно, в каждом контуре присутствует свой регулятор: РП – регулятор положения, РС – регулятор скорости, РТ – регулятор тока. ДПТ – двигатель постоянного тока типа 2ПБ160МГ (6 кВт), Р – выходной редуктор, ДП – датчик

положения, ДС – датчик скорости, ДТ – датчик тока, СП – силовой преобразователь.

Основным, интересующим нас регулятором, является ПИД-регулятор положения, отмеченный на структурной схеме аббревиатурой РП. Именно его параметры будут подвержены изменениям в процессе работы с целью адаптации управления к изменению параметров объекта.

Схема Simulink-модели следящей системы представлена на рис. 10.

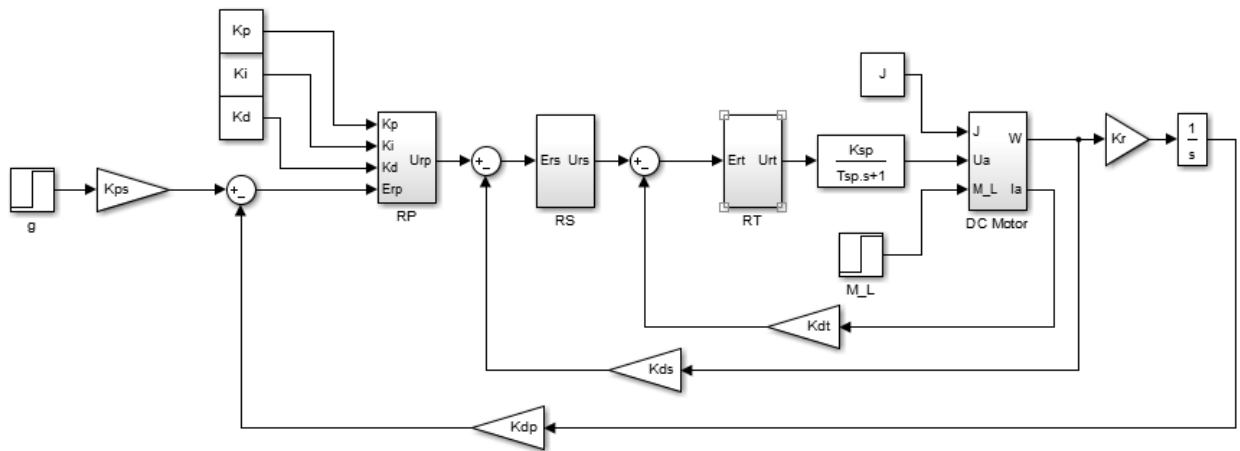


Рис. 10 Simulink-модель следящей системы

Построенная модель соответствует структуре, изображенной на рис. 9.

На рис. 10 представлена схема модели при проведении исследования динамических процессов в режиме позиционирования, о чем свидетельствует блок формирования ступенчатого задания  $g$ . При проведении исследований процессов в режиме слежения, входной ступенчатый сигнал заменяется непрерывным синусоидальным сигналом.

Далее подробно рассмотрим математическое описание каждого из элементов модели следящей системы.

## 2.2. Математическое описание следящей системы

### 2.2.1. Модель двигателя постоянного тока

На схеме модели следящей системы (рис. 10) двигатель постоянного тока независимого возбуждения представлен макроблоком DC Motor, внутренняя структура которого показана на рисунке 11.

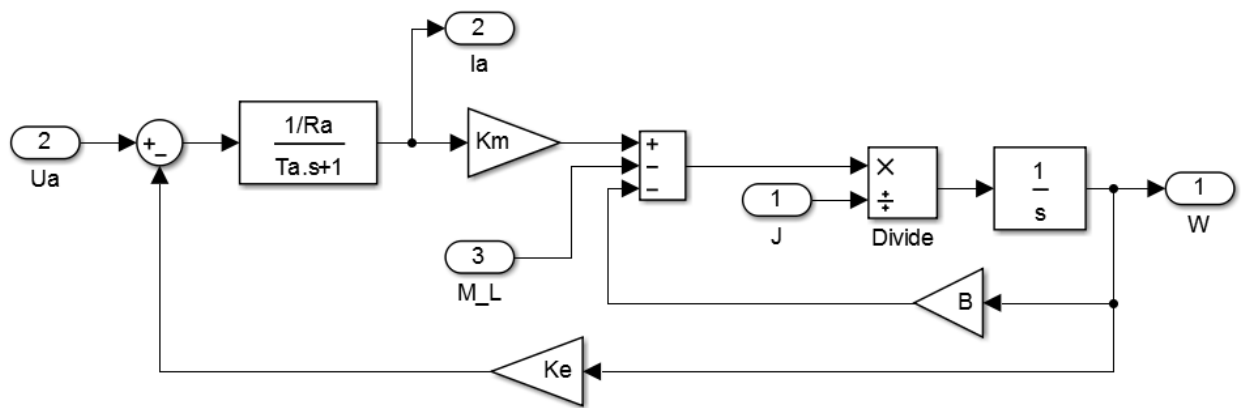


Рис. 11 Модель двигателя постоянного тока

Приведем паспортные данные двигателя для расчета параметров модели:

Тип: 2ПБ160МГ

$P=6000$  Вт - номинальная мощность двигателя

$U_{я.н.}=220$  В - номинальное напряжение якоря

$I_{я.н.}=30.7$  А - номинальный ток

$n_{д.н.}=2150$  об/мин - номинальная скорость вращения двигателя

$R_{я.ц.}=0.24$  Ом - сопротивление якорной цепи при 15 град Цельсия

$L_{я}=5.7$  мГн - индуктивность якоря

$M_{д.н.}=26.7$  Н\*м - номинальный электромагнитный момент двигателя

$J=0.0835$  кг\*м<sup>2</sup> - момент инерции якоря

$\dot{\omega}=320$  с<sup>(-2)</sup> - номинальное угловое ускорение двигателя

$T_{ДВ.э.} = 0.0234\text{с}$  - электромагнитная постоянная времени двигателя

По паспортным данным двигателя определим его дополнительные параметры.

Сопротивление двигателя, приведенное к 75 градусам Цельсия:

$$R_{ДВ} = R_{Я.Ц.}^{75^\circ} = R_{Я.Ц.}^{15^\circ} \cdot (1 + \alpha \Delta t) = 0.24 \cdot (1 + 0.004 \cdot 60) = 0.2976 \text{ Ом} \quad (2.1)$$

Номинальная скорость двигателя

$$\omega_H = \frac{\pi \cdot n_H}{30} = \frac{3.14 \cdot 2150}{30} = 225.033 \frac{\text{рад}}{\text{с}} \quad (2.2)$$

Конструктивные постоянные двигателя:

$$K_M = \frac{M_H}{I_H} = \frac{26.7}{30.7} = 0.87 \frac{\text{Н} \cdot \text{м}}{\text{А}} \quad (2.3)$$

$$K_E = \frac{U_H - I_H \cdot R_{ДВ}}{\omega_H} = \frac{220 - 30.7 \cdot 0.2976}{225.033} = 0.937 \frac{\text{В} \cdot \text{с}}{\text{рад}} \quad (2.4)$$

Электромагнитная постоянная времени двигателя:

$$T_a = \frac{L_a}{R_a} = 0.0234 \text{ с} \quad (2.5)$$

### 2.2.2. Силовой преобразователь

Силовой преобразователь (усилителя мощности) описывается апериодическим звеном первого порядка.

Примем основные параметры силового преобразователя (СП) равными:

$T_{СП} = 0.002 \text{ с}$  - постоянная времени апериодического звена, как модели СП.

$U_{ВХ.МАКС.} = 10 \text{ В}$  - максимальное напряжение на входе СП.

Вычислим расчетные параметры СП.

Максимальное напряжение на выходе СП:

$$U_{ВЫХ.МАКС} = 1.2 U_{Я.Н.} = 1.2 \cdot 220 = 264 \text{ В}; \quad (2.6)$$



Коэффициент усиления СП:

$$K_{SP} = \frac{U_{ВЫХ.МАКС.}}{U_{ВХ.МАКС.}} = \frac{264}{10} = 26,4; \quad (2.7)$$

Таким образом, зная параметры силового преобразователя можно построить его модель в виде апериодического звена с передаточной функцией:

$$W_{СП} = \frac{K_{SP}}{T_{SP} \cdot s + 1}; \quad (2.8)$$

### 2.2.3. Регуляторы

Регулятор контура тока Simulink-модели представлен макроблоком RT. Его структура представляет собой ПИ-регулятор с ограничением выходного сигнала и интегральной составляющей. Внутренняя структура блока RT изображена на рис. 12:

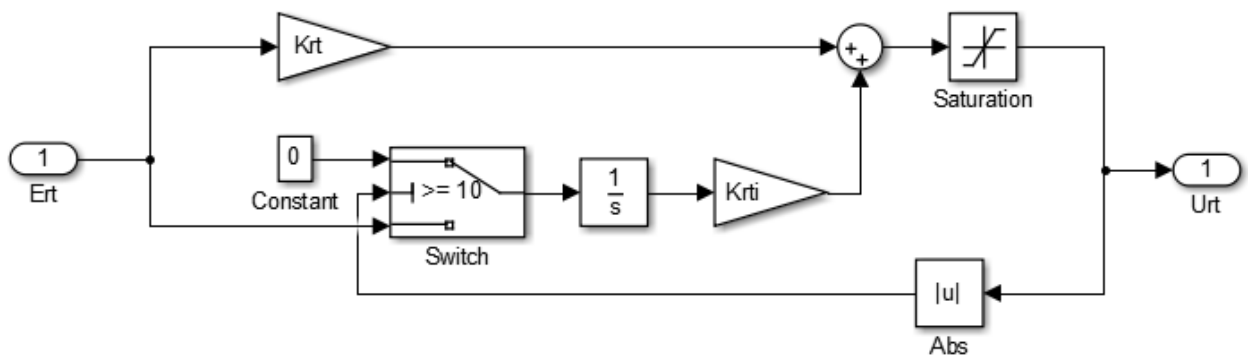


Рис. 12 Модель ПИ-регулятора тока RT

Блок Saturation служит для ограничения выходного сигнала регулятора в пределах  $\pm 10$ В. Блок Switch предназначен для того, чтобы при достижении верхнего или нижнего ограничения, прекращать процесс интегрирования.

Для того чтобы не увеличивать порядок системы, было принято решение использовать П-регулятор скорости, представленный макроблоком RS. Структура блока RS показана на рисунке 13:

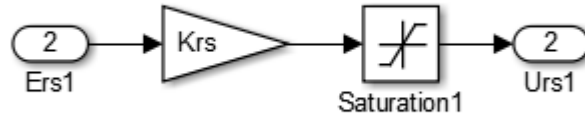


Рис. 13 Модель П-регулятора скорости RS

Модель ПИД-регулятора положения представлена блоком RP, структура которого изображена на рисунке 14. Коэффициенты усиления П, И и Д составляющих являются входными сигналами макроблока. Это сделано для того, чтобы была возможность применить алгоритмы адаптации параметров и менять их в процессе моделирования. Помимо коэффициентов усиления, в блок передаются значения рассогласования положения  $E_{rp}$  и текущего положения системы  $y$ . На вход дифференциальной составляющей регулятора подается текущее положение, вместо рассогласования, чтобы избежать больших скачков значений Д-составляющей.

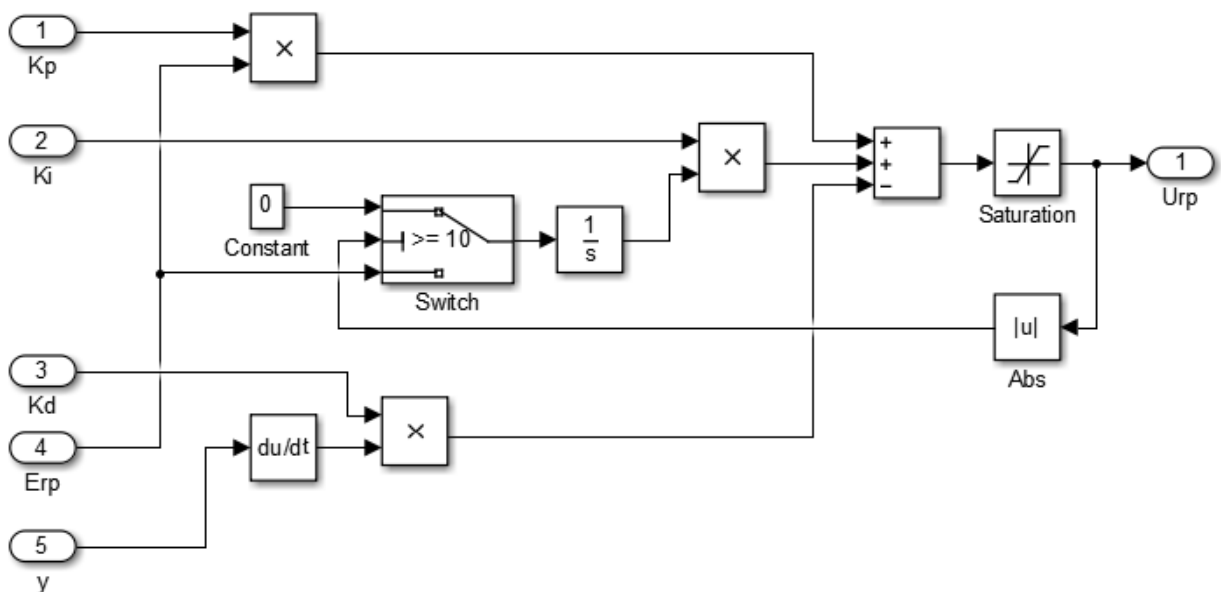


Рис. 14 Модель ПИД-регулятора

В параграфе 2.3 подробно описан процесс вычисления коэффициентов усиления регуляторов, здесь приведены лишь их итоговые значения.

Коэффициенты регулятора тока:

$$K_{RTP} = 0.243 ,$$

$$K_{RTI} = 10.382 ,$$

$$K_{RTD} = 0 .$$

Коэффициенты регулятора скорости:

$$K_{RSP} = 100$$

$$K_{RSI} = 0 ,$$

$$K_{RSD} = 0 .$$

Коэффициенты регулятора положения:

$$K_{RPP} = 50 ,$$

$$K_{RPI} = 500 ,$$

$$K_{RPD} = 0.2 .$$

#### **2.2.4. Модели датчиков и редуктора**

Датчики системы и редуктор представлены пропорциональными звеньями с различными коэффициентами передачи.

Редуктор представлен пропорциональным звеном с коэффициентом  $Kr = 0.2$ . После интегрирования значения выхода редуктора, получаем величину поворота вала двигателя  $\varphi$ .

Датчики тока, скорости и перемещения имеют следующие коэффициенты усиления:

$$K_{дт} = \frac{10}{1.2I_a} = \frac{10}{36} = 0.278 \frac{B}{A}; \quad (2.9)$$

$$K_{дс} = \frac{10}{1.2\omega} = \frac{10}{270} = 0.037 \frac{B}{с^{-1}}; \quad (2.10)$$

$$K_{дп} = \frac{10}{\varphi_{МАКС}} = \frac{10}{100} = 0.1 \frac{B}{рад}; \quad (2.11)$$

### 2.3. Настройка параметров регуляторов системы

Многоконтурные системы подчиненного управления находят широкое применение в системах управления электроприводами постоянного и переменного тока. В соответствии с общепринятыми принципами при настройке контуров параметры регуляторов выбираются таким образом, чтобы скомпенсировать большие инерционности объекта и привести передаточную функцию контура к нормированному виду, обусловленному выбранным критерием оптимизации.

Можно выделить два основных критерия оптимальной настройки контуров, применяемых на практике управления электроприводами:

- Модульный (технический) оптимум (МО)
- Симметричный оптимум (СО)

Данные методы настройки контуров обеспечивают переходные процессы, динамика которых чаще всего удовлетворяет требованиям систем управления электроприводами.

Контур тока спроектированной следящей системы настроен на модульный оптимум. Желаемая передаточная функция (ПФ) системы, настроенной на МО имеет вид:

$$W_{МО} = \frac{1}{2T_s(T_s + 1)} - \text{ПФ разомкнутой системы}, \quad (2.12)$$

$$\Phi_{MO} = \frac{1}{2T_s(T_s + 1)} - \text{ПФ замкнутой системы,} \quad (2.13)$$

где  $T$  – не скомпенсированная малая постоянная времени объекта.

Показатели качества переходного процесса замкнутого контура при данной настройке: перерегулирование  $\sigma=4.3\%$ , время первого согласования  $\tau=4.7T$ , время переходного процесса  $t_{\Pi}=8.4T$  (при трубке точности 2%).

На рисунке 14 изображена структурная схема контура тока и двигателя постоянного тока.

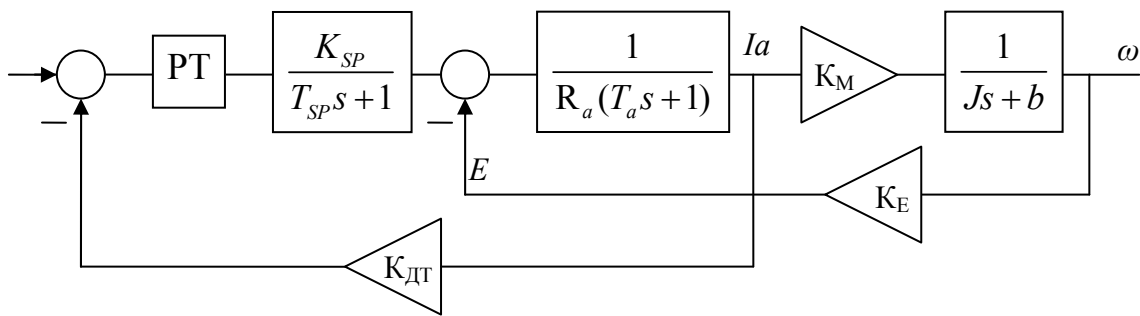


Рис. 15 Структурная схема контура тока и ДПТ

Составим передаточную функцию разомкнутого токового контура. Для упрощения данной процедуры, обратной связью по ЭДС двигателя можно пренебречь.

$$W_{OY\_TK} = W_{PT} W_{SP} W_{ДПТ} K_{ДТ} = W_{PT} \cdot \frac{K_{SP}}{T_{SP}s + 1} \cdot \frac{1}{R_a(T_a s + 1)} \cdot K_{ДТ}. \quad (2.14)$$

Коэффициент усиления датчика тока приняли как часть разомкнутой системы, чтобы замкнуть контур отрицательной обратной связью с коэффициентом передачи 1.

Для приведения данной передаточной функции к желаемой, необходимо выбрать следующую передаточную функцию регулятора тока:

$$W_{PT} = \frac{R_a(T_a s + 1)}{2T_{SP} K_{sp} K_{ДТ} s} \quad (2.15)$$

Коэффициенты передачи в модели при этом будут следующими:

$$Krt = \frac{T_a R_a}{2T_{SP} K_{sp} K_{дТ}} = 0.2392 \quad (2.16)$$

$$Trt = Ta = 0.0192 \quad (2.17)$$

$$Krti = \frac{Krt}{Trt} = 12.4586 \quad (2.18)$$

В этом случае ПФ контура тока принимает желаемый вид:

$$W_{КТ} = \frac{1}{2T_{SP}s(T_{SP}s + 1)} \text{ - ПФ разомкнутой системы,} \quad (2.19)$$

$$\Phi_{КТ} = \frac{1}{2T_{SP}^2 s^2 + 2T_{SP}s + 1} \text{ - ПФ замкнутого контура тока.} \quad (2.20)$$

Реакция на ступенчатое воздействие, приложенное к контуру тока, изображена на рисунке 15. Показатели процесса получились следующими: установившееся значение  $A=0.0273$ , перерегулирование  $\sigma=0.0312=14.3\%$ , время первого согласования  $\tau=0.0075=3.75T_{sp}$ , время переходного процесса  $t_{п}=0.045=22.5T_{sp}$ . Контур тока настроен на модульный оптимум, однако, показатели качества переходного процесса немного отличаются от желаемых из-за допущенного пренебрежения обратной связью по ЭДС двигателя.

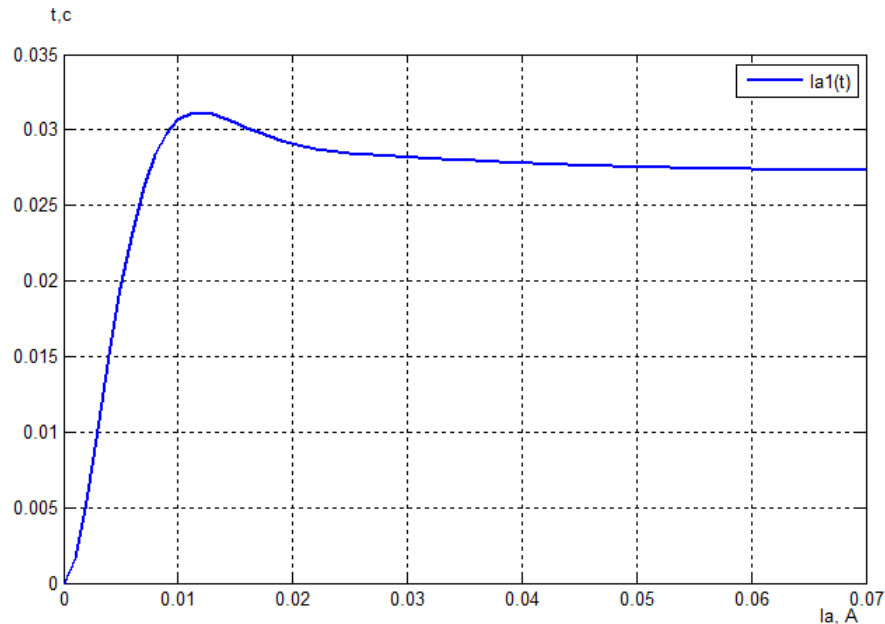


Рис. 16 Переходная характеристика контура тока при  $g=0.01$

Для того, чтобы не увеличивать порядок системы, в качестве регулятора скорости используется П-регулятор с коэффициентом усиления  $K_{RS} = 100$ .

График реакции контура скорости на ступенчатое воздействие изображен на рисунке 16. Показатели переходного процесса получились следующими: установившееся значение  $A=0.225$ , перерегулирование  $\sigma=0.2484=10.4\%$ , время первого согласования  $\tau=0.0136$  с, время переходного процесса  $t_{\Pi}=0.0379$ .

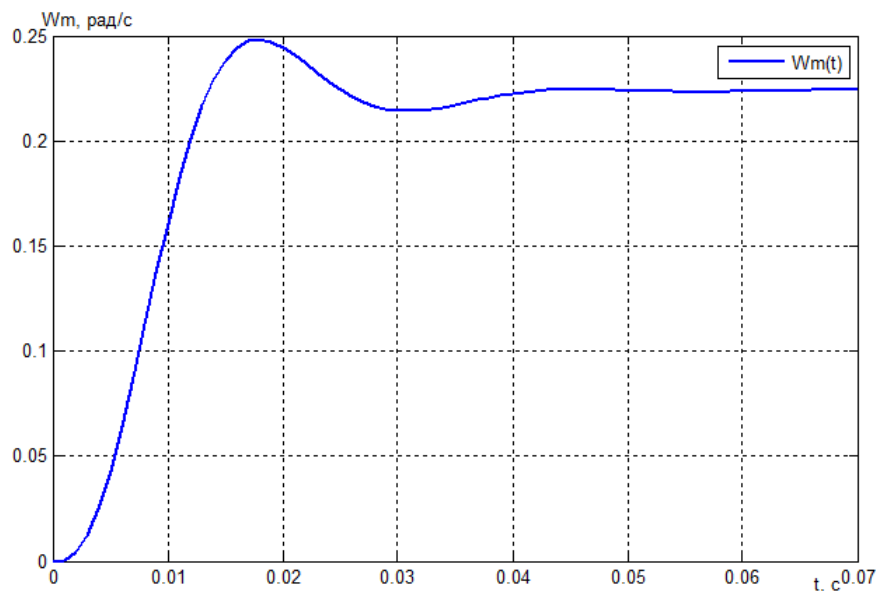


Рис. 17 Переходная характеристика контура скорости при  $g=0.01$

Параметры ПИД-регулятора положения подбирались исходя из стремления к максимальной скорости переходного процесса, отсутствию статической ошибки и минимальному перерегулированию. В качестве значений параметров регулятора прототипной системы были приняты следующие величины:

$$K_P = 50;$$

$$K_I = 500;$$

$$K_D = 0.2.$$

Переходный процесс следящей системы при ступенчатом воздействии представлен на рисунке 18. В момент времени  $t=0.7\text{c}$  подается момент нагрузки  $M_L = 15 \text{ Н*м}$ .

Анализируя переходный процесс прототипной системы, следует отметить наличие значительного перерегулирования, однако отработка задания и возмущающего воздействия происходит быстро и без статической ошибки.

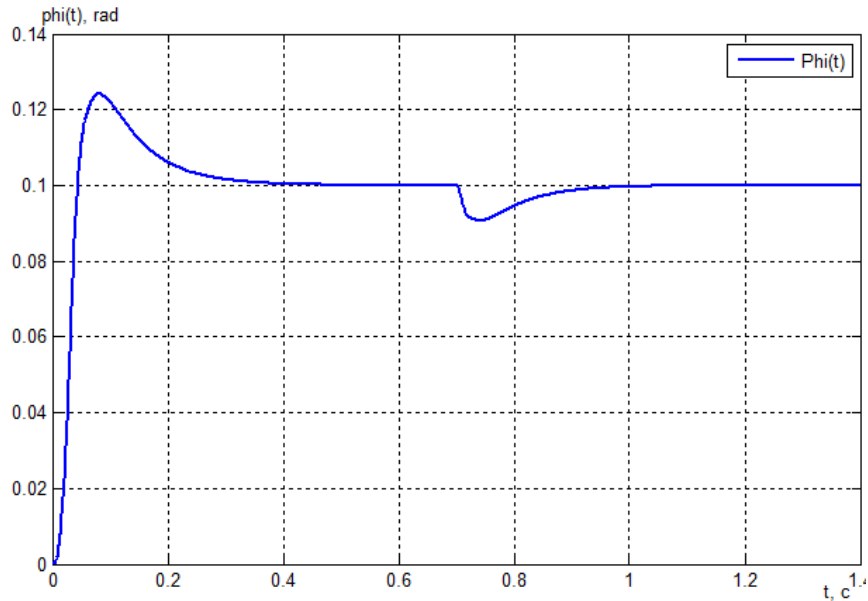


Рис. 18 Переходная характеристика системы при  $g=0.01$

## 2.4. Анализ динамических процессов системы

Процесс настройки параметров регуляторов системы проходил при фиксированных, известных заранее параметрах системы. Однако, на практике часто встречаются ситуации, когда параметры изменяются в процессе функционирования. Рассмотрим работу системы в режиме позиционирования и режиме слежения при разных значениях момента инерции на валу двигателя.



График изменения положения вала двигателя в режиме позиционирования при исходном значении момента инерции представлен на рисунке 18.

Работа системы в режиме слежения можно оценить, подавая в качестве задания периодический сигнал. Реакция системы на синусоидальный сигнал амплитудой  $g=0.01$  и частотой  $f=3$  рад/с при начальном значении момента инерции  $J=0.0835$  кг\*м<sup>2</sup> представлена на рис. 19, на котором для сравнения также изображен отмасштабированный сигнал задания.

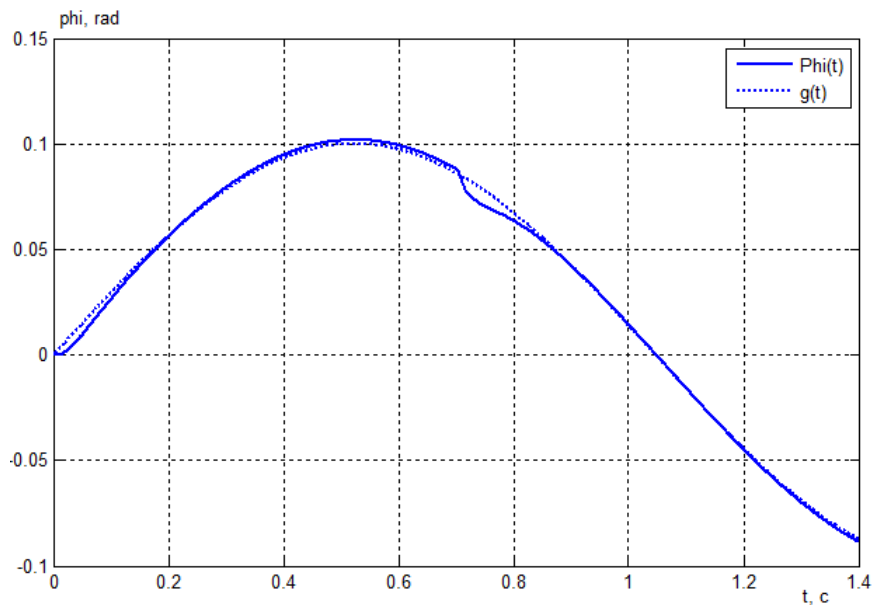


Рис. 19 Выход прототипной следящей системы в режиме слежения

Из графиков видно, что динамическая ошибка в режиме слежения практически отсутствует, за исключением периода начала движения и момента приложения возмущающего воздействия в виде момента нагрузки  $M_L=15$  Н\*м.

Рассмотрим работу системы, при моменте инерции в три раза большем начального значения  $3J=0.2505$  кг\*м<sup>2</sup> и в три раза меньшем  $J/3=0.0278$  кг\*м<sup>2</sup>. Результаты экспериментов представлены на графиках рис. 20 и 21.

При увеличенном моменте инерции на графике переходного процесса появилась выраженная колебательность с большим перерегулированием. Однако изменение момента инерции практически не повлияло на работу в

режиме слежения, на графике которого можно заметить лишь незначительное увеличение динамической ошибки в начальный момент времени и в момент приложения нагрузки.

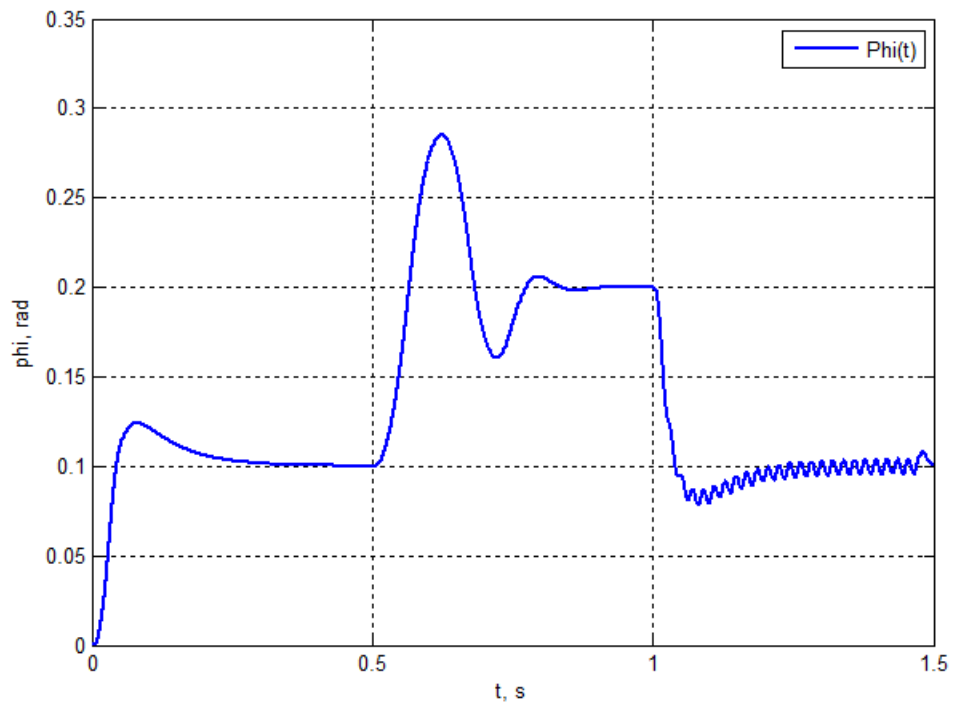


Рис. 20 Переходные процессы прототипной системы при изменяющемся моменте инерции

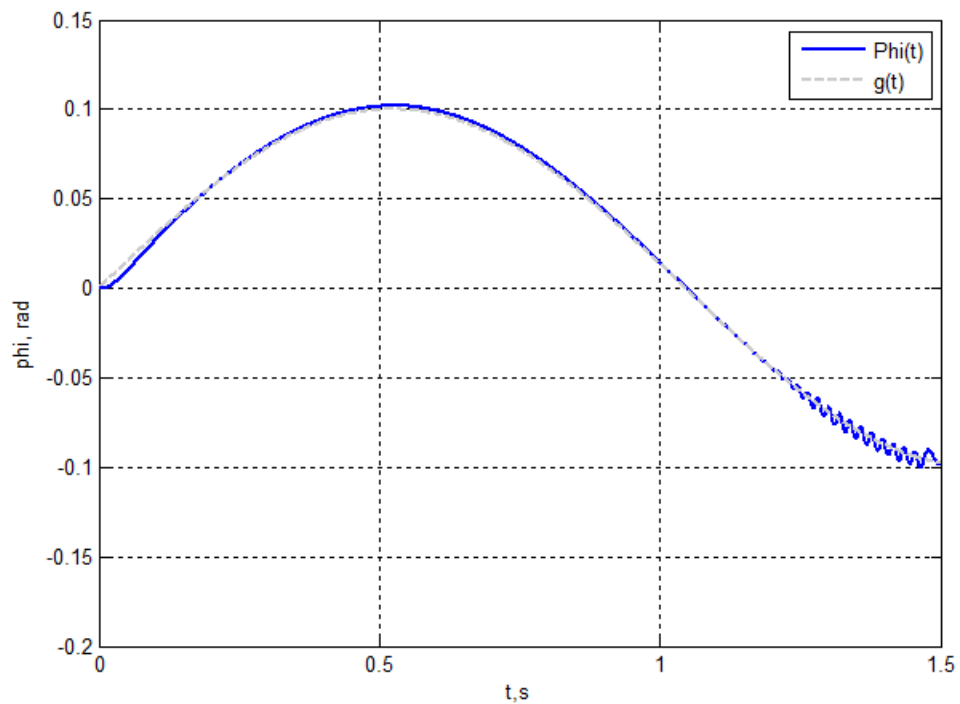


Рис. 21 Выход прототипной следящей системы в режиме слежения при изменяющемся моменте инерции

При уменьшенном в три раза моменте инерции на валу двигателя, на выходе системы появляются устойчивые колебания, и увеличивается динамическая ошибка в обоих режимах работы системы.

### **3. Разработка алгоритмов нейросетевой адаптации параметров ПИД-регулятора**

ПИД-регуляторы находят широкое применение в системах управления динамическими объектами, и в частности, в следящих системах. Тем не менее, применение классических ПИД-регуляторов имеет свои ограничения и недостатки. Как показали эксперименты, проведенные в п.2.4, при изменении параметров объекта значительно меняется качество управления. Для решения данной проблемы можно применить методы адаптации параметров регулятора.

В последние годы, разрабатываются и аппаратно реализуются регуляторы с автоматической адаптацией параметров. Но среди всех современных исследований и разработок в этой области, искусственные нейронные сети встречаются крайне редко. Это связано с необходимостью наличия больших вычислительных мощностей, особенно при наличии большого количества анализируемых параметров. Однако применение НС придают системам свойства интеллектуальности и расширяют возможности регулирования.

В связи с этим были разработаны два алгоритма, в основе которых лежит он-лайн обучение многослойных нейронных сетей, осуществляющих автоматическую адаптацию параметров регулятора в процессе функционирования следящей системы в условиях переменного момента инерции на валу двигателя.

В одном случае (п. 3.1–3.3) используется классический ПИД-регулятор с нейросетевой надстройкой, регулирующей его параметры. Во втором случае (п. 3.4–3.6) классический регулятор полностью заменяется нейросетевым регулятором, параметры которого меняются в зависимости от параметров системы.

### 3.1. Модификация следящей системы блоком нейросетевой адаптации

Первым из двух реализованных методов адаптации параметров регулятора положения рассмотрим метод гибридного нейро-ПИД управления. Данный метод подразумевает совместное функционирование классического ПИД-регулятора и нейросетевого модуля, который реализует алгоритм параметрической адаптации. Дополним разработанную Simulink-модель модулем нейросетевой адаптации. Функциональная схема системы представлена на рисунке 22.

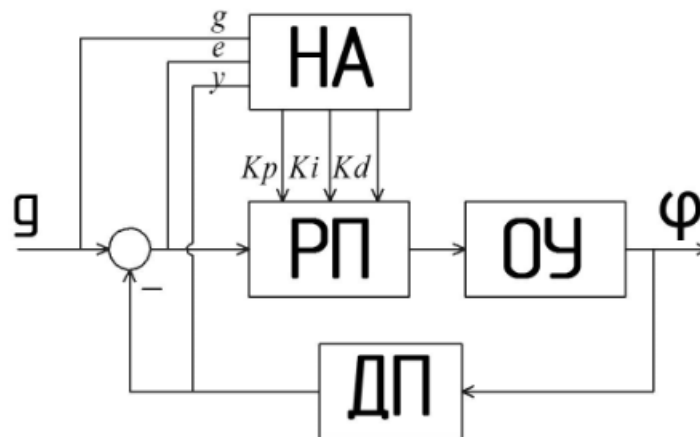


Рис. 22 Функциональная схема следящей системы с гибридным нейро-ПИД управлением

На схеме рис.22 изображен контур положения следящей системы, где НА- блок, выполняющий нейросетевой алгоритм, РП – регулятор положения, ДП – датчик положения, ОУ – объект управления, содержащий в себе контур тока и скорости ДПТ, а также понижающий редуктор.

Алгоритм адаптации обрабатывает текущую информацию о состоянии ОУ с помощью трехслойной нейронной сети и выдает значения для коэффициентов регулятора для улучшения качества регулирования. Входными параметрами блока НА являются текущее входное воздействие следящей системы  $g$ , рассогласование  $e$  и выходная величина системы  $y$ .

Выходными величинами блока НА являются значения коэффициентов усиления  $K_p$ ,  $K_i$  и  $K_d$  ПИД-регулятора.

### 3.2. Математическое описание блока нейросетевой адаптации

В основе алгоритма блока адаптации параметров регулятора лежит трехслойная нейронная сеть прямого распространения, изменение весовых коэффициентов которой происходит по градиентному методу обратного распространения ошибки. Структура нейронной сети представлена на рисунке 23:

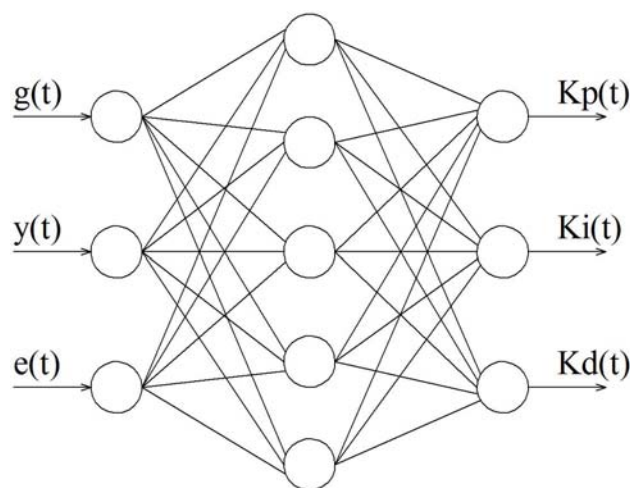


Рис. 23 Структурная схема нейронной сети в основе блока НА

Взаимодействие искусственной нейронной сети с обратным распространением ошибки и традиционного ПИД-регулятора может эффективно решать задачи, связанные с недостаточной определенностью параметров ПИД-управления. Это объясняется возможностью НС аппроксимировать любую нелинейную функцию, простотой структуры сети и особенностями алгоритма обучения.

Схема алгоритма работы блока нейросетевой адаптации представлена на рисунке 24:

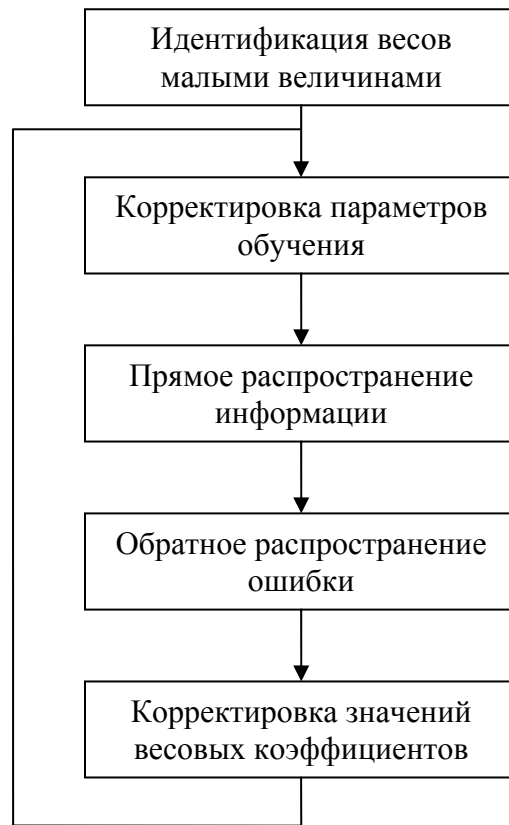


Рис. 24 Алгоритм работы блока НА

На первом шаге происходит идентификация весов малыми случайными величинами в диапазоне  $(-0.25; 0.25)$ . Скорость обучения принимается равной  $\eta = 1$ .

В зависимости от рассогласования в системе, изменяется параметр скорости обучения  $\eta$ . Пока присутствует ошибка и если она увеличивается, то коэффициент скорости обучения умножается на понижающий коэффициент 0.9. В случае, если ошибка уменьшается, то скорость обучения умножается на повышающий коэффициент 1.2.

После этого происходит прямое распространение сигналов от входного слоя к выходному. Информация, поступившая на входной слой на  $k$ -й итерации, передается всем нейронам среднего слоя без изменений:

$$\begin{cases} out_1^{(1)}(k) = in_1^{(1)}(k) = g(k), \\ out_2^{(1)}(k) = in_2^{(1)}(k) = y(k), \\ out_3^{(1)}(k) = in_3^{(1)}(k) = e(k), \end{cases} \quad (3.1)$$

где  $i=1..3$  – номер нейрона входного слоя,  $in_i^1$  – значение, поступившее на вход  $i$ -го нейрона первого слоя,  $out_i^1$  – выход нейрона.

Математические операции, выполняемые в скрытом слое на  $k$ -й итерации:

$$in_j^{(2)}(k) = \sum_{i=0}^m w_{ji}(k) \cdot out_i^{(1)}(k); \quad (3.2)$$

$$out_j^{(2)}(k) = f(in_j^{(2)}(k)), \quad (3.3)$$

где  $m=3$  – число нейронов входного слоя,  $j=1..5$  – номер нейрона скрытого слоя,  $in_j^2$  – значение, полученное на входе функции активации  $j$ -го нейрона,  $w_{ji}$  – синаптический вес связи  $j$ -го нейрона скрытого слоя и  $i$ -го нейрона входного слоя, при этом,  $w_{j0}$  (соответствующий весу связи для фиксированного сигнала  $out_0^1=1$ ) равен порогу  $b_j$ .  $out_j^2$  – выход  $j$ -го нейрона скрытого слоя,  $f$  – сигмоидальная функция активации нейрона, равная:

$$f(x) = \tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}; \quad (3.4)$$

Аналогичные операции выполняются и в выходном слое сети:

$$in_l^{(3)}(k) = \sum_{j=0}^n w_{lj}(k) \cdot out_j^{(2)}, \quad (3.5)$$

$$out_l^{(3)}(k) = g(in_l^{(3)}(k)); \quad (3.6)$$

где  $n = 5$  – число нейронов скрытого слоя,  $l=1..3$  – номер нейрона выходного слоя, остальные обозначения аналогичны обозначениям для скрытого слоя.



Т.к. коэффициенты регулятора не могут принимать отрицательные значения, то функция активации нейронов выходного слоя принята следующей:

$$g(x) = \frac{1 + \tanh(x)}{2} = \frac{e^x}{e^x + e^{-x}}. \quad (3.7)$$

После прямого распространения информации, происходит обратное распространение ошибки с целью минимизации критерия обучения:

$$E = \frac{1}{2}(g(k) - y(k))^2; \quad (3.8)$$

Алгоритм обратного распространения состоит в применении к синаптическим весам  $w$  коррекции  $\Delta w$ , пропорциональной частной производной  $\frac{\partial E(k)}{\partial w(k)}$ .

$$\Delta w_{ij}(k) = -\eta \frac{\partial E(k)}{\partial w_{ij}(k)} + \alpha \Delta w_{ij}(k-1), \quad (3.9)$$

где  $\eta$  - скорость обучения,  $\alpha$  - коэффициент инерции.

Рассмотрим вычисление коррекция  $\Delta w_{ij}$  для нейронов выходного слоя.

Градиент можно представить в следующем виде:

$$\frac{\partial E(k)}{\partial w_{ij}(k)} = \frac{\partial E(k)}{\partial y(k)} \cdot \frac{\partial y(k)}{\partial \Delta u(k)} \cdot \frac{\partial \Delta u(k)}{\partial out_l(k)} \cdot \frac{\partial out_l(k)}{\partial in_l(k)} \cdot \frac{\partial in_l(k)}{\partial w_{ij}(k)}, \quad (3.10)$$

где  $\Delta u(k)$  - приращение значения выхода регулятора положения на  $k$ -й итерации.

Рассмотрим отдельные множители выражения для градиента. Дифференцируя обе части выражения (3.5) по  $\partial w_{ij}(k)$  получим:

$$\frac{\partial in_l(k)}{\partial w_{ij}(k)} = out_j^{(2)}. \quad (3.11)$$

Дифференцируя обе части выражения (3.6) по  $\partial in_l(k)$  получим:

$$\frac{\partial out_l(k)}{\partial in_l(k)} = g_l'(in_l). \quad (3.12)$$

В свою очередь

$$g_l'(in_l) = 2 \cdot g(in_l) \cdot (1 - g(in_l)). \quad (3.13)$$

Дискретное представление уравнения ПИД-контроллера может быть представлено как:

$$\Delta u(k) = K_p \cdot [e(k) - e(k-1)] + K_I \cdot e(k) + K_D \cdot [e(k) - 2e(k-1) + e(k-2)],$$

следовательно:

$$\begin{cases} \frac{\partial \Delta u(k)}{\partial out_1(k)} = e(k) - e(k-1); \\ \frac{\partial \Delta u(k)}{\partial out_2(k)} = e(k); \\ \frac{\partial \Delta u(k)}{\partial out_3(k)} = e(k) - 2e(k-1) + e(k-2); \end{cases} \quad (3.14)$$

Так как  $\frac{\partial y(k)}{\partial \Delta u(k)}$  неизвестно, то следует заменить его на  $\operatorname{sgn}\left(\frac{\partial y(k)}{\partial \Delta u(k)}\right)$ .

Дифференцируя обе части выражения (3.8) по  $\partial y(k)$  получим:

$$\frac{\partial E_{lj}(k)}{\partial y(k)} = y. \quad (3.15)$$

В результате можно записать:

$$\Delta w_{lj}(k) = -\eta \cdot \delta_l(k) \cdot out_j^{(2)} + \alpha \cdot \Delta w_{lj}(k-1), \quad (3.16)$$

где  $\delta_l(k)$  - локальный градиент нейрона l.

Или в развернутом виде:

$$\Delta w_{1j}(k) = -\eta \cdot y(k) \cdot \operatorname{sgn}\left(\frac{\partial y(k)}{\partial \Delta u(k)}\right) \cdot (e(k) - e(k-1)) \cdot 2 \cdot g(in^{(3)}_1) \cdot (1 - g(in^{(3)}_1)) \cdot out_j + \alpha \Delta w_{1j}(k-1) \quad (3.17)$$

$$\Delta w_{2j}(k) = -\eta \cdot y(k) \cdot \operatorname{sgn}\left(\frac{\partial y(k)}{\partial \Delta u(k)}\right) \cdot e(k) \cdot 2 \cdot g(in^{(3)}_2) \cdot (1 - g(in^{(3)}_2)) \cdot out_j + \alpha \Delta w_{2j}(k-1) \quad (3.18)$$

$$\Delta w_{3j}(k) = -\eta \cdot y(k) \cdot \operatorname{sgn}\left(\frac{\partial y(k)}{\partial \Delta u(k)}\right) \cdot \frac{\partial \Delta u(k)}{\partial out_3(k)} \cdot 2 \cdot g(in^{(3)}_3) \cdot (1 - g(in^{(3)}_3)) \cdot out_j + \alpha \Delta w_{3j}(k-1) \quad (3.19)$$

Вычисления приращений весов для нейронов скрытого слоя отличается от вычислений для нейронов выходного слоя. Сигнал ошибки для каждого

нейрона скрытого слоя рекурсивно вычисляется на основе ошибок всех нейронов, с которыми он непосредственно связан. Общий вид выражения для приращений остается неизменным:

$$\Delta w_{ji}(k) = -\eta \cdot \delta_j(k) \cdot \text{out}_i^{(1)} + \alpha \cdot \Delta w_{ji}(k-1) \quad (3.20)$$

Однако меняется определение локальных градиентов. Для нейронов скрытого слоя локальный градиент равен произведению производной функции активации  $f_j'(in_j)$  на взвешенную сумму градиентов, вычисленных для нейронов следующего (выходного) слоя.

$$\delta_j(k) = f_j'(in_j) \cdot \sum_l \delta_l(k) \cdot w_{lj}(k) \quad (3.21)$$

После этого, на основании вычисленных приращений, можно произвести коррекцию синаптических весов для всех нейронов сети.

$$\begin{aligned} w_{lj}(k) &= w_{lj}(k-1) + \Delta w_{lj}(k) \\ w_{ji}(k) &= w_{ji}(k-1) + \Delta w_{ji}(k) \end{aligned} \quad (3.22)$$

Затем, когда все весовые коэффициенты скорректированы, снова проходит прямое распространение сигналов от входов НС к выходам, и т.д. За счет постоянного корректирования параметров обучения и весовых коэффициентов, нейро-ПИД регулятор адаптируется к изменениям параметров ОУ, основываясь на измеренных значениях показателей состояния системы.

### 3.3. Моделирование следящей системы с нейро-ПИД управлением

На рисунке 25 представлена Simulink-модель следящей системы с гибридным нейро-ПИД управлением.

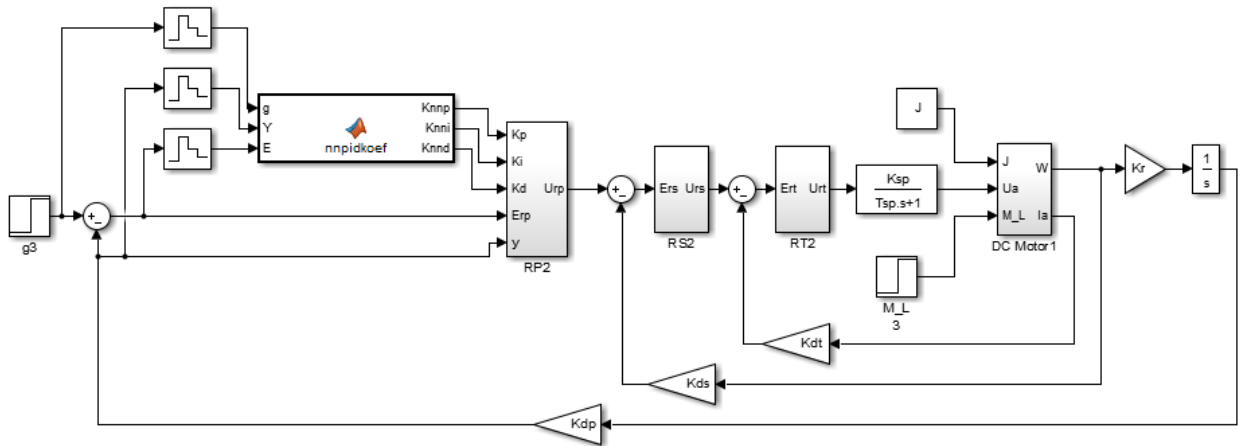


Рис. 25 Simulink-модель следящей системы с нейро-ПИД управлением

Пользовательский блок `nnpidcoef`, изображенный на рисунке 26 реализует алгоритм НА, описанный в п.3.1-3.2. На его вход поступают задание  $g$ , рассогласование  $e$  и выход системы  $y$ , пропущенные через задержки нулевого порядка. Выходы блока представляют собой коэффициенты регулятора положения.

Ниже представлены результаты работы системы. На Рис. 26 показан график реакции системы на ступенчатые воздействия. При этом, перед подачей очередной ступени, в системе меняется момент инерции. Изначально он равен  $J = 0.0835 \text{ кг}\cdot\text{м}^2$ , он увеличивается в 3 раза  $3J = 0.2505 \text{ кг}\cdot\text{м}^2$  и перед последним заданием, уменьшается в три раза относительно исходной величины  $J/3 = 0.0278 \text{ кг}\cdot\text{м}^2$ . На Рис. 28 изображен график выходной величины при подаче на вход системы синусоидального задания. Условия изменения момента те же самые. Время изменения 0,5 и 1 секунда.

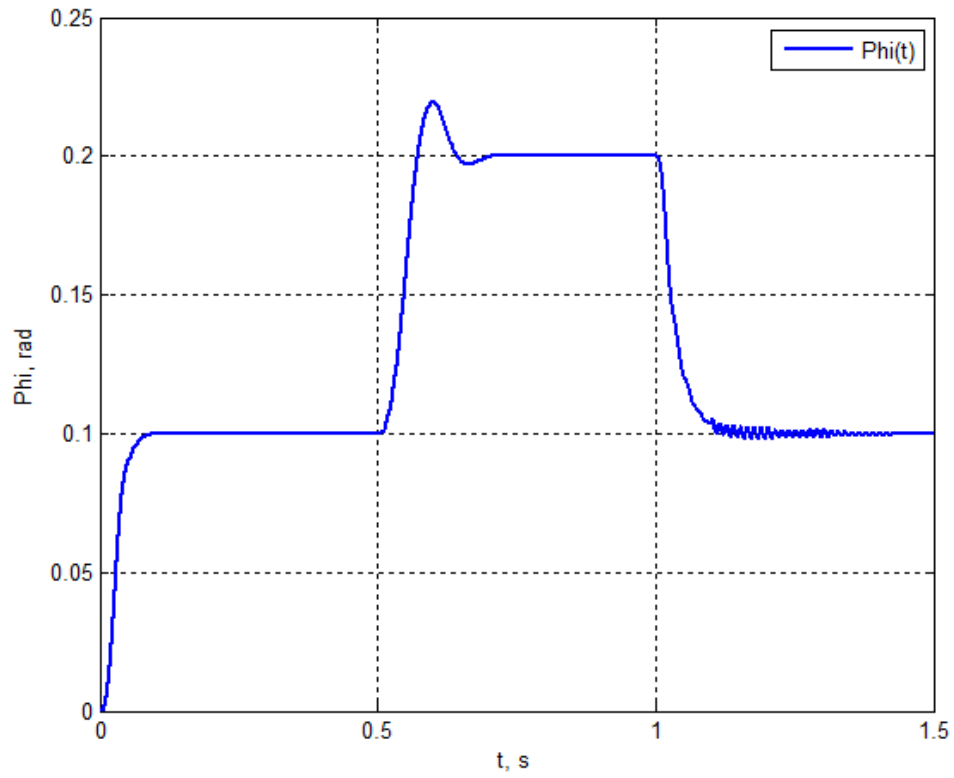


Рис. 26 Реакция системы с нейро-ПИД регулятором на ступенчатые воздействия при изменяющемся моменте инерции

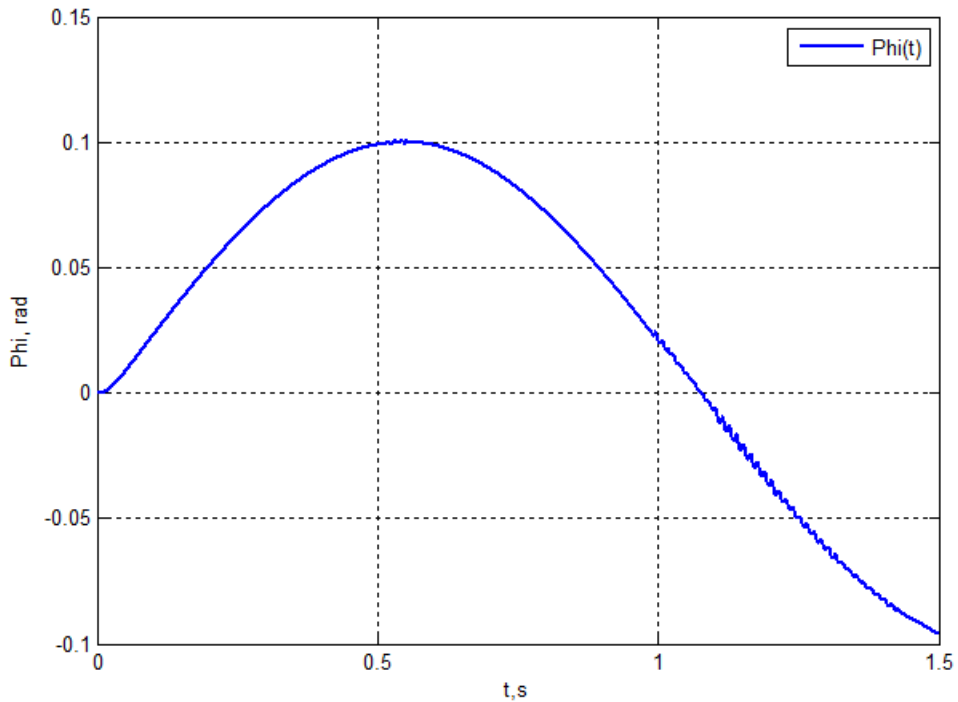


Рис. 27 Реакция системы с нейро-ПИД регулятором на синусоидальное воздействие при изменяющемся моменте инерции

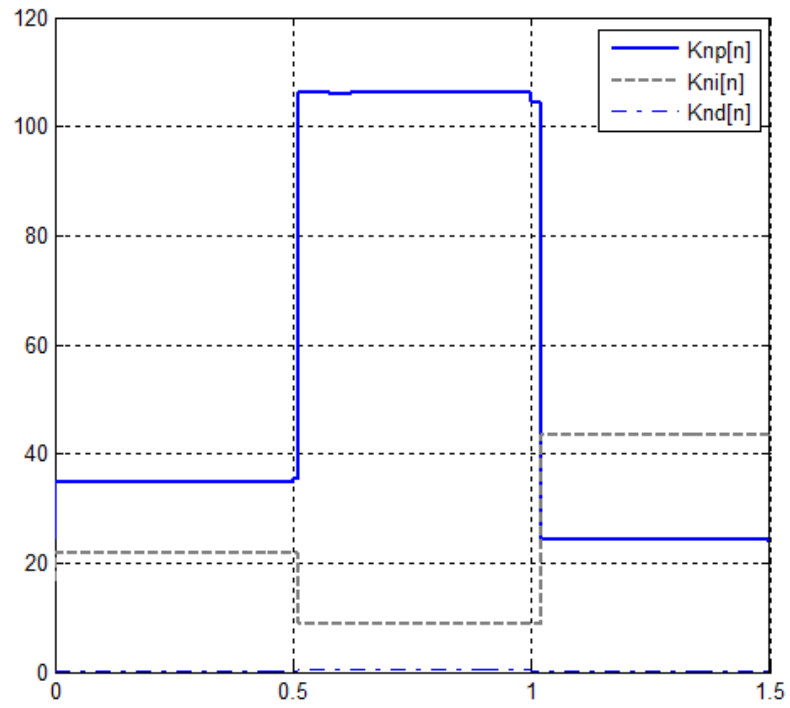


Рис. 28 График изменения параметров регулятора

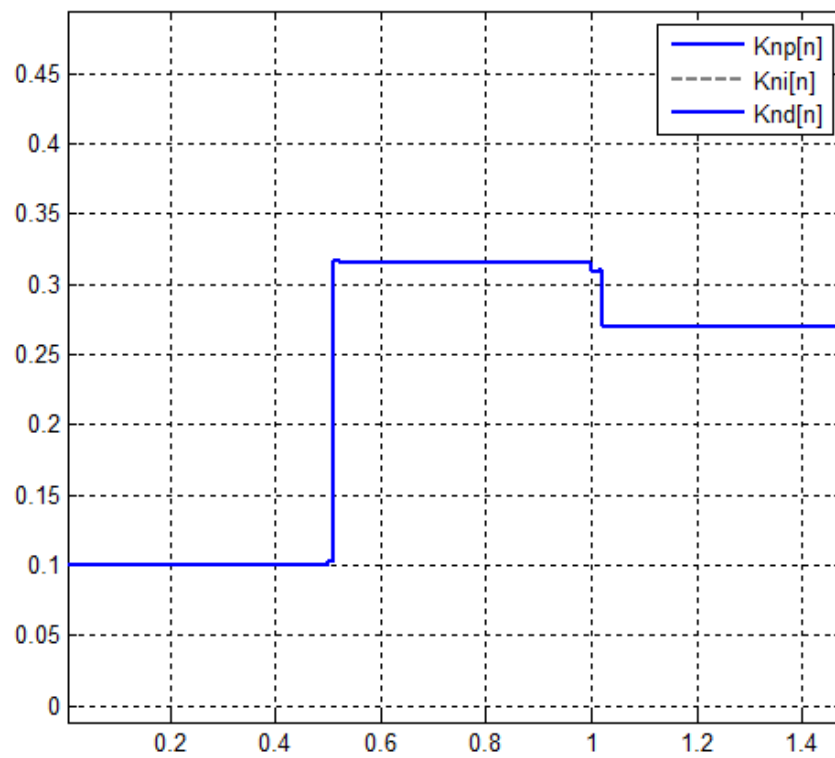


Рис. 29 График изменения коэффициента усиления дифференциальной составляющей

### 3.4. Замена ПИД-регулятора на нейросетевой регулятор

Другим методом адаптации параметров, рассмотренным в данной работе является метод близкий к подражающему нейроуправлению. Подражающее нейроуправление подразумевает офф-лайн обучение нейросетевого контроллера на примере существующего классического регулятора с последующей его заменой в контуре управления. В отличие от описанного выше метода, реализованный способ нейросетевого управления обучается он-лайн на поступающей на его входы информации о состоянии системы.. Заменяем в прототипной Simulink-модели ПИД-регулятор положения на нейросетевой регулятор. Функциональная схема системы представлена на рисунке 30.

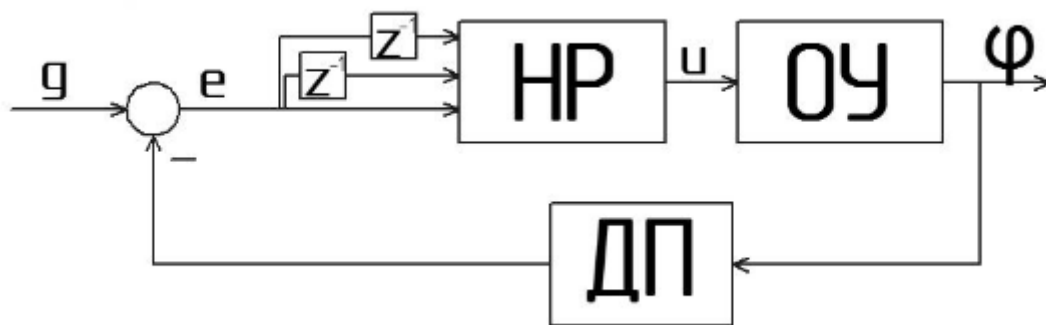


Рис. 30 Функциональная схема следящей системы с нейроконтроллерным управлением

Аналогично рисунку 22, на функциональной схеме рис. 29 представлен контур положения следящей системы. НР – нейросетевой регулятор, ДП – датчик положения, ОУ - объект управления, содержащий в себе контур тока и скорости ДПТ, а также понижающий редуктор.

В основе адаптивного регулятора положения НР лежит трехслойная нейронная сеть прямого распространения. Входными сигналами для блока (и для нейронной сети) являются сигнал задания  $g$  и сигнал обратной связи  $u$ .

Выходным сигналом блока НР является управляющее воздействие на объект управления  $u$ .

Адаптивность достигается путем он-лайн обучения НС на данных о состоянии системы, поступающих на регулятор.

### 3.5. Математическое описание нейросетевого регулятора

В основе алгоритма блока адаптивного нейросетевого регулятора лежит трехслойная нейронная сеть прямого распространения, структуру которой можно описать как (3-3-1). Изменение синаптических весов, так же как и в предыдущем случае, происходит по градиентному методу обратного распространения ошибки. Структура нейронной сети представлена на рисунке 31.

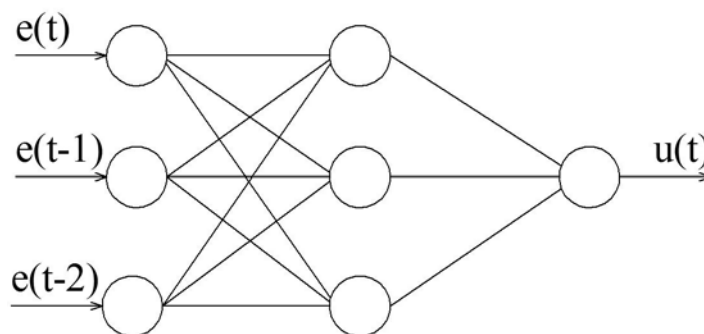


Рис. 31 Структурная схема нейронной сети в основе блока НР

В отличие от предыдущего случая, данная нейронная сеть непосредственно выполняет функции регулятора. Весовые коэффициенты синаптических связей между нейронами скрытого и выходного слоев, настраивающиеся в процессе обучения во время функционирования системы, играют роль коэффициентов усиления регулятора положения.

Блок-схема алгоритма работы блока НР полностью совпадает со схемой алгоритма блока НА, представленной на рисунке 25.



При запуске модели происходит идентификация весов малыми случайными величинами в диапазоне  $(-0.5; 0.5)$ .

После этого происходит прямое распространение сигналов от входного слоя к выходному. Информация, поступившая на входной слой на  $k$ -й итерации, передается всем нейронам среднего слоя без изменений:

$$\begin{cases} out_1^{(1)}(k) = in_1^{(1)}(k) = e(k), \\ out_2^{(1)}(k) = in_2^{(1)}(k) = e(k-1), \\ out_3^{(1)}(k) = in_3^{(1)}(k) = e(k-2), \end{cases} \quad (3.23)$$

где  $i=1..3$  – номер нейрона входного слоя,  $in_i^1$  – значение, поступившее на вход  $i$ -го нейрона первого слоя,  $out_i^1$  – выход нейрона.

Математические операции, выполняемые в скрытом слое на  $k$ -й итерации:

$$in_j^{(2)}(k) = \sum_{i=1}^m w_{ji}(k) \cdot out_i^{(1)}(k); \quad (3.24)$$

$$out_j^{(2)}(k) = f(in_j^{(2)}(k)), \quad (3.25)$$

где  $m=3$  – число нейронов входного слоя,  $j=1..3$  – номер нейрона скрытого слоя,  $in_j^2$  – значение, полученное на входе функции активации  $j$ -го нейрона,  $w_{ji}$  – синаптический вес связи  $j$ -го нейрона скрытого слоя и  $i$ -го нейрона входного слоя,  $f$  – сигмоидальная функция активации нейрона, равная:

$$f(x) = \frac{1}{1 + e^{-x}}; \quad (3.25)$$

Аналогичные операции выполняются и в выходном слое сети:

$$in^{(3)}(k) = \sum_{j=1}^n w_j(k) \cdot out_j^{(2)}, \quad (3.26)$$

$$out^{(3)}(k) = g(in^{(3)}(k)); \quad (3.27)$$

где  $n = 3$  – число нейронов скрытого слоя, остальные обозначения аналогичны обозначениям для скрытого слоя.

Функция активации нейронов выходного слоя принята такой же, как и для скрытого слоя:

$$g(x) = \frac{1}{1 + e^{-x}}. \quad (3.28)$$

После прямого распространения информации, происходит обратное распространение ошибки с целью минимизации критерия обучения:

$$E = \frac{1}{2}(g(k) - y(k))^2 = \frac{1}{2}e(k)^2; \quad (3.29)$$

Алгоритм обратного распространения состоит в применении к синаптическим весам  $w$  коррекции  $\Delta w$ , пропорциональной частной производной  $\frac{\partial E(k)}{\partial w(k)}$ .

$$\Delta w_j(k) = -\eta \frac{\partial E(k)}{\partial w_j(k)}, \quad (3.30)$$

где  $\eta$  - скорость обучения.

Рассмотрим вычисление коррекция  $\Delta w_j$  для нейронов выходного слоя.

Градиент можно представить в следующем виде:

$$\frac{\partial E(k)}{\partial w_j(k)} = \frac{\partial E(k)}{\partial e(k)} \cdot \frac{\partial e(k)}{\partial \Delta u(k)} \cdot \frac{\partial \Delta u(k)}{\partial out(k)} \cdot \frac{\partial out(k)}{\partial in(k)} \cdot \frac{\partial in(k)}{\partial w_j(k)}, \quad (3.31)$$

Выполнив операции, аналогичные рассмотренным в п.3.1, получим:

$$\frac{\partial E(k)}{\partial w_j(k)} = e(k) \cdot \frac{\partial e(k)}{\partial \Delta u(k)} \cdot (-1) \cdot g(in) \cdot (1 - g(in)) \cdot out_j^{(2)} = -1 \cdot \frac{\partial e(k)}{\partial \Delta u(k)} \cdot \delta^1 \cdot out_j \quad (3.32)$$

$$\text{где } \delta^1 = e(k) \cdot g(in) \cdot (1 - g(in)) \quad (3.33)$$

Множителем  $\frac{\partial e(k)}{\partial \Delta u(k)}$  пренебрежем, т.к. он неизвестен и его значение

можно скомпенсировать в параметре скорости обучения  $\eta$ . В результате можно записать:

$$\Delta w_j(k) = -\eta \cdot \delta^1(k) \cdot \text{out}_j^{(2)}, \quad (3.34)$$

Вычисления приращений весов для нейронов скрытого слоя отличается от вычислений для нейронов выходного слоя. Сигнал ошибки для каждого нейрона скрытого слоя рекурсивно вычисляется на основе ошибки нейрона выходного слоя. Общий вид выражения для приращений остается неизменным:

$$\Delta w_{ji}(k) = -\eta \cdot \delta^2(k) \cdot \text{out}_i^{(1)} \quad (3.35)$$

Однако меняется определение локальных градиентов. Для нейронов скрытого слоя локальный градиент равен произведению производной функции активации  $f'(in_j)$  на взвешенный градиент нейрона следующего (выходного) слоя.

$$\delta^2(k) = f'(in_j) \cdot (1 - f(in_j)) \cdot \delta^1(k) \cdot w_j(k) \quad (3.36)$$

После этого, на основании вычисленных приращений, можно произвести коррекцию синаптических весов для всех нейронов сети.

$$\begin{aligned} w_j(k) &= w_j(k-1) + \eta \cdot \delta^1(k) \cdot \text{out}_j^{(2)} \\ w_{ji}(k) &= w_{ji}(k-1) + \eta \cdot \delta^2(k) \cdot \text{out}_i^{(1)} \end{aligned} \quad (3.37)$$

Затем, когда все весовые коэффициенты скорректированы, снова проходит прямое распространение сигналов от входов НС к выходам, и т.д. За счет постоянного корректирования параметра обучения и весовых коэффициентов, нейро-ПИД регулятор адаптируется к изменениям параметров ОУ, основываясь на измеренных значениях показателей рассогласования.

### 3.6. Моделирование следящей системы с нейрорегулятором

На рис.32 представлена модель системы с нейросетевым регулятором.

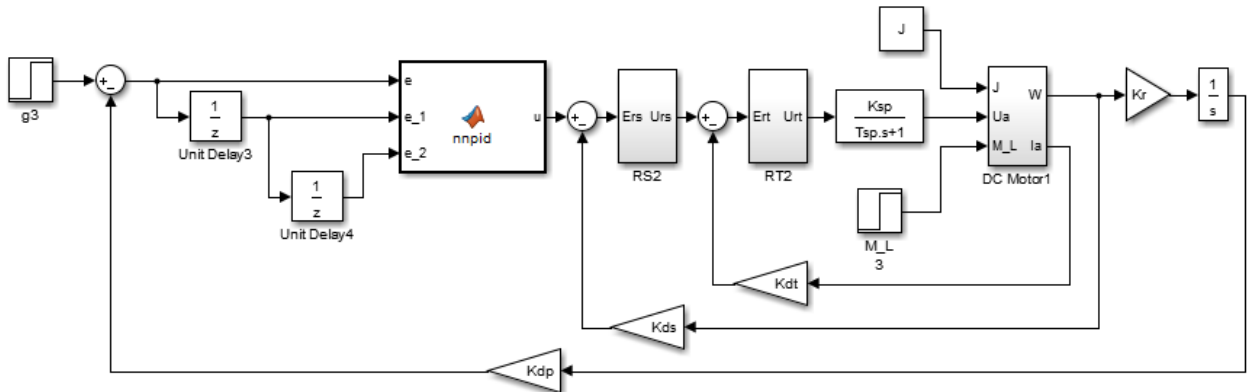


Рис. 32 Simulink-модель следящей системы с нейросетевым регулятором

Пользовательский блок `nnpid`, изображенный на рисунке 32 реализует алгоритм НР, описанный в п.3.5. На его вход поступают рассогласование  $e$ , рассогласование с задержкой на один таймерный цикл и с задержкой на два цикла. Выход блока представляет собой управляющее воздействие на контур положения.

Ниже представлены результаты работы системы. На Рис. 33 показан график реакции системы на ступенчатые воздействия. При этом, перед подачей очередной ступени, в системе меняется момент инерции. Изначально он равен  $J = 0.0835 \text{ кг}\cdot\text{м}^2$ , он увеличивается в 3 раза  $3J = 0.2505 \text{ кг}\cdot\text{м}^2$  и перед последним заданием, уменьшается в три раза относительно исходной величины  $J/3 = 0.0278 \text{ кг}\cdot\text{м}^2$ . На Рис. 34 изображен график выходной величины при подаче на вход системы синусоидального задания. Условия изменения момента те же самые. Время изменения 0,5 и 1 секунда.

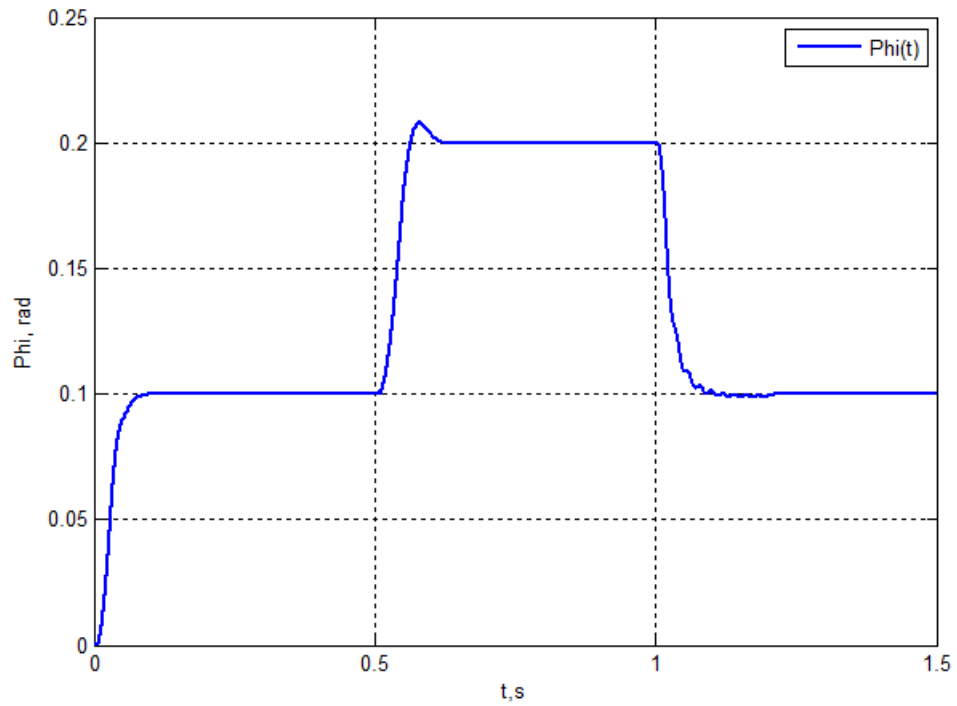


Рис. 33 Реакция системы с нейросетевым регулятором на ступенчатые воздействия при изменяющемся моменте инерции

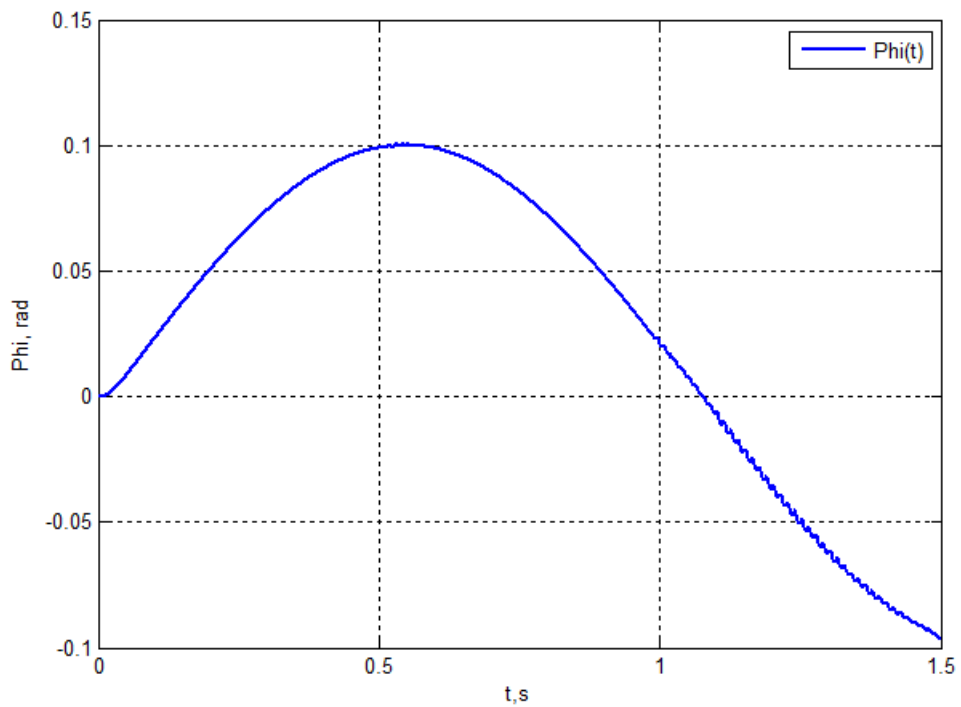


Рис. 34 Реакция системы с нейросетевым регулятором на синусоидальное воздействие при изменяющемся моменте инерции

Сравнивая результаты работы системы при двух разных способах нейросетевого управления, можно сделать вывод о том, что оба способа значительно улучшают качество регулирования, относительно традиционного метода управления с ПИД-регулятором.

В случае с гибридным нейро-ПИД управлением, адаптация системы к изменению момента инерции происходит быстро, но немного менее качественно, чем в случае с нейрорегулятором. Об этом можно судить по величине перерегулирования при увеличенном моменте инерции и амплитуде колебаний при уменьшенной инерционности.

## 4. Вопросы технической реализации

### 4.1 Инструменты программной реализации нейровычислений

В последнее время области применения искусственных нейронных сетей стремительно расширяются, в связи с этим для реализации необходимых вычислений производится все больше программных и аппаратных продуктов, специализирующихся на нейровычислениях.

Следует отметить, большинство реализаций нейронных сетей и нейровычислений представляет собой программные решения. Самым распространенным программным пакетом для изучения и моделирования НС является Neural Network Toolbox в среде Matlab фирмы Mathworks.

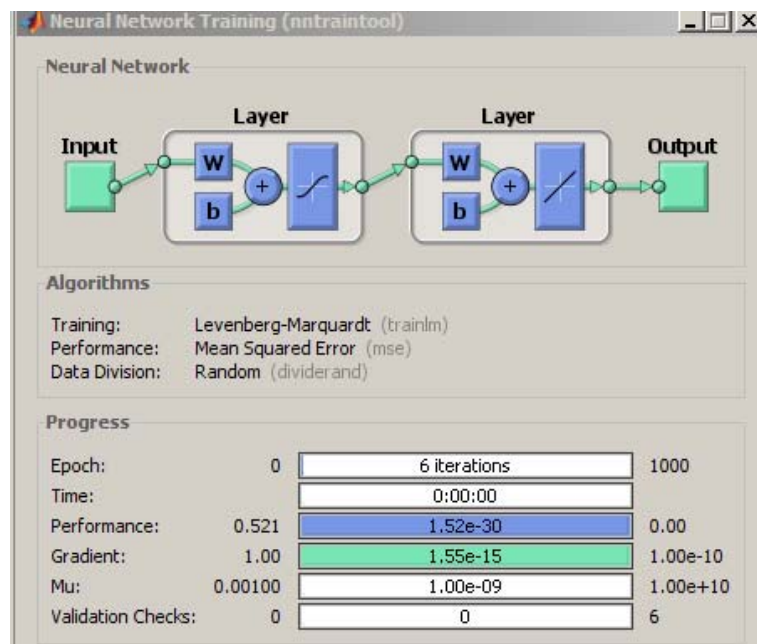


Рис. 35 Представление НС и процесса обучения в Neural Network Toolbox.

Данный пакет имеет большую базу примеров создания и обучения НС, позволяющую получить представление о топологиях, методах обучения и применении НС, что делает его подходящим для пользователей, не имеющих опыта работы с НС.

Созданные в Neural Network Toolbox нейронные сети могут быть использованы как для анализа накопленных данных (прогнозирование, кластеризация) так и для обработки информации, поступающей в режиме реального времени с каких-либо измерительных устройств или из других программ. Также для обеспечения необходимой вычислительной мощности в случае с большими наборами данных имеется возможность распределить вычисления в случае наличия многоядерных процессоров или компьютерных кластеров средствами Matlab.

Одним из самых передовых и эффективных нейросетевых программных продуктов является STATISTICA Automated Neural Networks фирмы Statsoft.

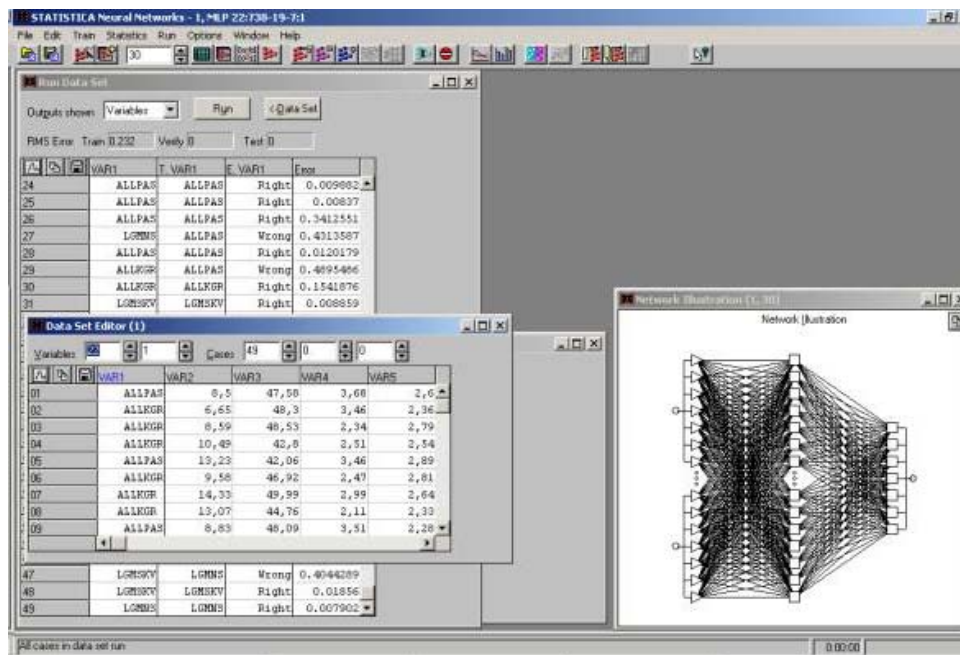


Рис. 36 Представление НС и анализаторов данных в программе STATISTICA

STATISTICA имеет широкий спектр возможностей, нацеленный на экспертов в области нейросетевых вычислений. Вместе с тем, для пользователей, не имеющих большого опыта в работе с НС, в данном пакете существует инструмент автоматизированного нейросетевого поиска Автоматизированная нейронная сеть. Этот инструмент проводит пользователя через все этапы создания различных нейронных сетей и выбирает наиболее подходящую для решения поставленных задач.



Так же, как и Matlab Neural Network Toolbox, STATISTICA может работать с архивами данных или с активными данными, несущими информацию о технологических процессах. Важным достоинством является наличие генератора исходного кода созданной обученной нейронной сети, что позволяет интегрировать ее в пользовательские приложения или в программу контроллера.

Так, например, после создания и обучения нейронной сети для применения в управлении следящей системой, можно сгенерировать исходный код на желаемом языке программирования. На основе этого кода имеется возможность создать пользовательский функциональный блок в среде программирования «Полигон» фирмы Промавтоматка. Полученный блок, реализующий функции нейронной сети, можно добавлять в управляющую программу контроллера, являющуюся программным регулятором положения. Однако, действуя подобным образом, нельзя забывать о технических возможностях используемого оборудования. Если в результате разработки получилась НС с большим числом нейронов, множеством синаптических связей и большими объемами обрабатываемой информации, то качественное регулирование в масштабе реального времени будет невозможным при отсутствии больших вычислительных мощностей.

## **4.2 Аппаратные средства реализации нейровычислений**

В настоящее время большинство реализаций нейронных сетей представляют собой программные продукты. Это объясняется тем, что НС сейчас активно развиваются, и попытки различные экспериментальные методы обучения и топологии гораздо проще с использованием ПО, т.к. создание ПО более гибкий процесс, чем создание аппаратного продукта. Тем не менее, у аппаратной реализации НС есть ряд преимуществ перед программными продуктами.

Основным преимуществом является скорость обработки информации, увеличение которой достигается путем проведения параллельных вычислений. К тому же, при использовании отдельного модуля, выполняющего нейровычисления, снижается нагрузка на основной процессор и выполнение других приложений не задерживается.

Другим немаловажным преимуществом является стоимость использования аппаратного нейросетевого модуля при необходимости обработки большого количества данных. Отдельный специализированный модуль стоит дешевле, чем центральный процессор с такой же вычислительной мощностью.

Также следует отметить, что применение аппаратной реализации вычислений приводит к увеличению надежности системы обработки данных, по сравнению со случаем использования программной модели, за счет меньшей вероятности отказа оборудования.

Как уже было сказано, на сегодняшний день нейрокомпьютеры и нейрочипы не часто встречаются в использовании, ввиду удобства использования программных моделей. Если нейронная сеть обрабатывает не большой поток данных и имеет несложную структуру, как в нашем случае, то ее вполне можно реализовать программно, как часть управляющей программы программируемого контроллера. Единственным, на что нужно обратить внимание – это на количество времени, затрачиваемое на выполнение всех вычислений в НС, и хватает ли времени в таймерном цикле на обработку всей программы. Иными словами, необходим контроллер с тактовой частотой, достаточной для выполнения всех операций в программе.

В качестве примера такого контроллера приведем контроллер Beckhoff CX2020 (рис.37), широко используемый в промышленных системах автоматизации. Контроллер содержит 1.4 GHz Intel Celeron CPU в качестве центрального процессора, что позволяет ему выполнять сложные управляющие программы за короткий таймерный цикл. Поэтому добавление разработанных нейросетевых алгоритмов не скажется на работе

управляющей программы и качество управления в реальном времени сохранится. Существенным недостатком является стоимость, которая составляет примерно \$3500.

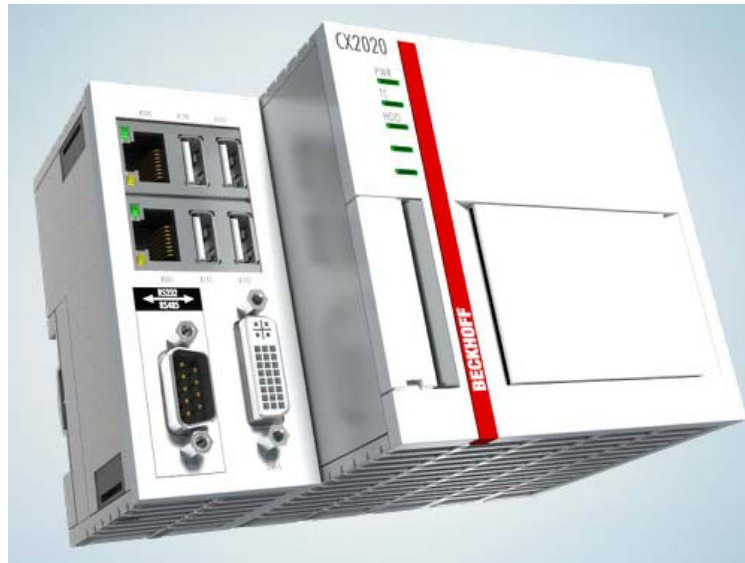


Рис. 37 Внешний вид контроллера Beckhoff CX2020

При реализации НС с большим объемом вычислений, все же следует прибегнуть к специализированным нейрочипам, которые сейчас можно свободно найти в продаже. В качестве примера можно привести чип CM1K фирмы The CogniMem Technologies.

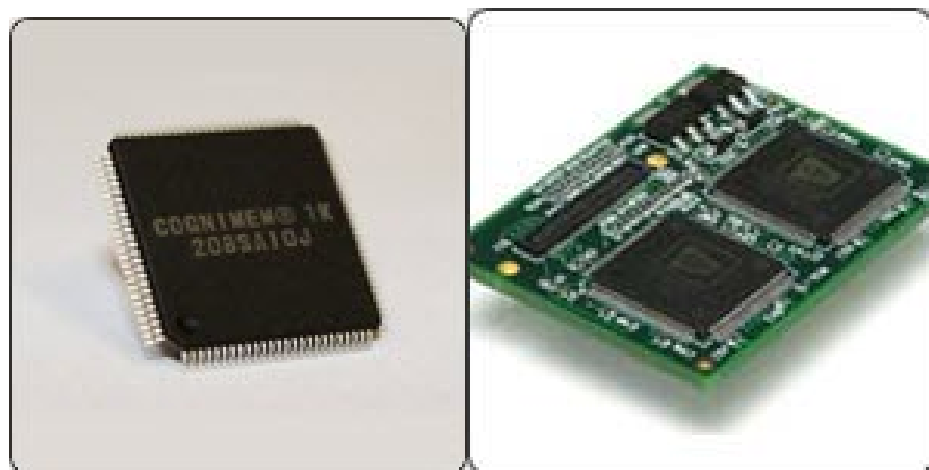


Рис. 38 Внешний вид нейрочипа CM1K и модуля CM1K

Данный чип содержит 1024 нейрона, работающих параллельно. При необходимости увеличения количества нейронов в сети, имеется возможность приобрести модуль СМЕ1К с двумя нейрочипами СМК1. Также рассмотренные чипы позволяют нарастить сеть до 8192 нейронов, путем соединения четырех таких модулей. Связь с ними осуществляется по последовательной шине I2C (400 кбит/с). Примерная стоимость \$97.

Выбор способа технической реализации нейросетевого управления зависит от текущего состояния системы управления. Если система управляется с помощью программной модели на ПК, то достаточно использовать один из указанных программных пакетов, для создания программного модуля, реализующего нужный алгоритм.

В случае, если система управляется с помощью программируемого контроллера, и имеется среда программирования с возможностью добавления пользовательских подпрограмм, то дешевле всего будет добавить сгенерировать исходный код разработанного алгоритмы и добавить в качестве подпрограммы в контроллер.

Если же такой возможности нет, или контроллер не имеет достаточной вычислительной мощности, то необходимо вводить дополнительный нейросетевой модуль, информацию с которого использовать при реализации закона управления.

## Заключение

В результате теоретических и экспериментальных исследований работы разработаны алгоритмы нейросетевой адаптации параметров ПИД-регулятора и адаптивного нейросетевого регулятора. Проведено моделирование следящей системы с изменяющимся во времени моментом инерции, для проверки свойств адаптации системы и качества регулирования.

Оценив результаты моделирования системы, можно сделать вывод о работоспособности реализованных алгоритмов и пригодности их использования для управления следящей системой в условиях нестационарности момента инерции на валу двигателя.

Выявлено, что немного лучше с задачей адаптации справляется нейросетевой регулятор, о чем свидетельствуют графики изменения выходной величины системы (угла поворота двигателя). Однако преимущество этого метода оказалось не очень существенно, применительно к исследуемой системе.

Также рассмотрены вопросы технической реализации разработанных алгоритмов и приведены рекомендации по выбору способа осуществления. Основным фактором при выборе является текущее техническое состояние системы управления, т.к. от этого существенно зависит цена вопроса.

## Список сокращений

ДПТ – двигатель постоянного тока

ИНС (НС) - искусственная нейронная сеть

ИСУ – интеллектуальная система управления

МО – модульный оптимум

НА – нейросетевой алгоритм

НР - нейросетевой регулятор

ПК – персональный компьютер

ПО – программное обеспечение

ПФ – передаточная функция

САУ – система автоматического управления

СП – силовой преобразователь

## Список литературы

1. Бесекерский В.А., Попов Е.П. Теория систем автоматического управления. – М.: Москва, 1975. -770 с.
2. Еременко Ю.И., Полещенко Д.А., Глущенко А.И. Об интеллектуальной адаптации параметров пид-регулятора для снижения энергопотребления управляемого процесса// Научные ведомости БелГУ. - 2013. № 22 (165). Выпуск 28/1
3. Короткий С. Нейронные сети. Алгоритм обратного распространения
4. Короткий С. Нейронные сети. Обучение без учителя
5. Круглов В.В., Борисов В. В. Искусственные нейронные сети. Теория и практика. -2-е изд., стереотип. - М.: Горячая линия-Телеком, 2002. - 382 с: ил.
6. Михайлов А.С. Актуальные проблемы применения нейрорегуляторов в системах автоматического управления
7. Терехов С.А. Технологические аспекты обучения нейросетевых машин. // Лекция для VIII Всероссийской научно-технической конференции “Нейроинформатика -2006”
8. Фролов С.В., Елизаров И.А., Солуданов М.Н. Самонастраивающаяся нейросетевая система управления //Вестник ТГТУ. - 2005. Том 11. № 4.
9. Хайкин С. Нейронные сети: полный курс, 2-е изд., испр. : Пер. с англ. –М. : ООО «И.Д. Вильямс», 2006. – 1104 с. : ил. Парал. Тит. Англ.
10. Чернодуб А.Н., Дзюба Д.А. Обзор методов нейроуправления // Проблемы программирования. -2011. № 2. с. 79-94.
11. Cui S.G, Pan H.L. Application of Self-tuning of PID Control Based on BP Neural Networks in the Mobile Robot Target Tracking // Third International Conference on Instrumentation, Measurement, Computer, Communication and Control. -2013

12. Ji H., Li Z. Design of Neural Network PID Controller Based on Brushless DC Motor// Second International Conference on Intelligent Computation Technology and Automation. -2009
13. Kan J., Liu J. Self-Tuning PID Controller Based on Improved BP Neural Network // Second International Conference on Intelligent Computation Technology and Automation. -2009
14. Ponce A.N., Behar A.A., Hernández A.O. Neural Networks for Self-tuning Control Systems // Czech Technical University in Prague. Acta Polytechnica Vol. 44 No. 1/2004
15. Song Shoujun and Liu Weiguo (2011). Application of Improved PID Controller in Motor Drive System, PID Control, Implementation and Tuning, Dr. Tamer Mansour (Ed.), ISBN: 978-953-307-166-4, InTech, Availablefrom: <http://www.intechopen.com/books/pid-control-implementation-and-tuning/application-of-improved-pidcontroller-in-motor-drive-system>