

УДК 004.89 + 681.51
ББК 32.813

О РАЗРАБОТКЕ МЕТОДА ВЫБОРА СТРУКТУРЫ НЕЙРОННОЙ СЕТИ ДЛЯ РЕШЕНИЯ ЗАДАЧИ АДАПТАЦИИ ПАРАМЕТРОВ ЛИНЕЙНЫХ РЕГУЛЯТОРОВ

Еременко Ю. И.¹, Глущенко А. И.²

*(Старооскольский технологический институт
им. А.А. Угарова (филиал) ФГАОУ ВО НИТУ «МИСиС»,
Старый Оскол)*

Рассмотрена проблема выбора структуры нейронной сети (НС), осуществляющей в составе нейросетевого оптимизатора настройку параметров П-, ПИ- или ПИД-регулятора в реальном времени при управлении нелинейными объектами. Предложен подход, позволяющий без построения модели объекта управления (ОУ) определять количество слоев сети и нейронов в них, функции активации и время задержки сигналов выхода ОУ, подаваемых на вход НС. Разработан алгоритм построения и инициализации НС нейросетевого оптимизатора. Результаты численных экспериментов на моделях ОУ и практической реализации для двух электронагревательных печей показали экономию времени и энергоносителя на выполнение графика заданий при использовании оптимизатора по сравнению с обычным линейным регулятором, что подтверждает адекватность предложенного подхода.

¹ Юрий Иванович Еременко, доктор технических наук, профессор (erem49@mail.ru).

² Антон Игоревич Глущенко, кандидат технических наук, доцент (г. Старый Оскол, м-н Макаренко, д. 42, тел. (4725) 45-12-17, strondutt@mail.ru).

Ключевые слова: нейронная сеть, адаптивное управление, ПИД-регулятор, нейросетевой оптимизатор, выбор структуры нейронной сети, время задержки входных сигналов.

1. Введение

В промышленности значительная часть контуров управления технологическими процессами функционирует на основе ПИД-алгоритма регулирования [23]. Данный закон управления обладает рядом преимуществ, таких как простота понимания и реализации и высокая надежность. В то же время он является линейным, а реальные объекты управления (ОУ) в большинстве случаев нелинейны. Подобное несоответствие приводит к тому, что коэффициенты регулятора, подобранные при первичной настройке для конкретного режима работы ОУ, не позволяют получать сходных по качеству переходных процессов в других режимах. Это ведет к увеличению времени на отслеживание уставок и расхода энергоносителя.

Примером таких объектов являются нагревательные ОУ – печи нагрева литой заготовки, дуговые печи и пр. Изменение задания, графика уставок, степени загрузки печи при постоянных коэффициентах ПИД-регулятора зачастую приводит к снижению качества регулирования и к росту энергозатрат на ведение технологических процессов, поскольку для подобных ОУ принудительное охлаждение невозможно, ввиду чего наличие даже небольшого перерегулирования требует длительного времени на охлаждение [59].

Теоретически для каждого состояния ОУ необходимо подбирать свои значения коэффициентов регулятора. Однако в реалиях производства чаще всего используют всего один набор таких коэффициентов для всех режимов и не производят их перенастройку ввиду трудоемкости данного процесса.

Для решения указанной проблемы необходимо построение адаптивных или оптимальных систем управления технологическим процессом. Одним из наиболее проработанных подходов к построению оптимальных регуляторов является методика аналитического конструирования оптимальных регуляторов

(АКОР) [1, 8, 9, 11, 12, 14, 16, 17, 50]. Конечная цель АКОР состоит в получении закона управления аналитическим путем, исходя из предъявляемых к качеству управления требований, формализованных в форме интегрального критерия качества. Теория АКОР достаточно проработана применительно, в первую очередь, к линейным объектам и квадратичным функционалам-критериям качества [9, 12, 14]. Применение указанных подходов для синтеза управляющих систем для нелинейных объектов приводит к нелинейным краевым задачам. Решение подобных задач представляет собой выполнение сложных вычислительных процедур [14]. Кроме того, достаточно сложно решаемой остается проблема выбора весовых коэффициентов для минимизируемого функционала оптимальности. И, что не менее важно, не устанавливается функциональная связь между реальными показателями качества управления и весовыми коэффициентами функционала оптимальности, что приводит к многократному повторению процедуры синтеза регулятора, основанному на знании модели объекта управления, для достижения требуемых показателей переходных процессов [14]. Кроме того, построение оптимального регулятора [1, 8, 9, 11, 12, 14, 16, 17, 50] предполагает отказ от классической структуры ПИД-регулятора, что, ввиду вышесказанного, повышает сложность внедрения подобных разработок на производстве.

Таким образом, более перспективным подходом представляется проведение оперативной адаптации параметров существующего ПИД-регулятора, являющегося привычным для работников производства. Для настройки указанных регуляторов разработано достаточно большое число методов, обзор которых подробно приведен в [24, 55, 56]. В целом среди них можно выделить группу классических методов, таких как:

1) метод Циглера–Никольса [80] (требуется либо наличие модели ОУ, либо вывод системы управления на границу устойчивости), в большинстве случаев позволяющий получать коэффициенты, значения которых достаточно далеки от оптимальных;

2) методы, основанные на построении модели ОУ с помощью специального испытательного сигнала (ступенчатого [18,

31, 53, 59, 63] или полигармонического [15, 19]): метод частотной идентификации и настройки А.Г. Александрова [19], метод, положенный в основу работы адаптивных ПИД-регуляторов фирмы Siemens [59] (на подобном же принципе основаны адаптивные регуляторы фирм Omron, Control Station и др.) и пр.

Эти методы предполагают полный отказ от используемых в настоящее время в регуляторе коэффициентов и подбор новых. Моменты времени, когда необходимо использовать метод адаптации, определяются человеком, как и амплитуда испытательного сигнала. Таким образом, применение данных методов для автоматической оперативной подстройки параметров регуляторов является затруднительным.

Кроме того, метод, описанный в [19], сталкивается с определенными сложностями при идентификации ОУ с постоянной времени более 800 с, а для методов, основанных на применении ступенчатого тестового сигнала, в частности [59], результаты идентификации ОУ даже в сходных условиях могут весьма существенно отличаться. В целом же построение модели реального нелинейного объекта управления является весьма нетривиальной и трудноосуществимой в реалиях производства задач.

В то же время человек, занимающийся на производстве настройкой линейных регуляторов, способен осуществлять их подстройку в реальном масштабе времени без тестовых сигналов и модели. Это возможно благодаря его знаниям, опыту, способностям к обучению, обобщению и прогнозированию. Указанные характеристики в определенной мере позволяют воспроизводить интеллектуальные методы, что делает перспективным их применение для рассматриваемой проблемы.

Методы роя частиц [20] и генетических алгоритмов [27, 70, 71, 75] применяются для решения задач оптимизации, причем изначально случайным образом генерируется исходное множество решений. В случае с задачей настройки параметров регулятора, каждое решение – это набор значений параметров K_p , K_I , K_D . В качестве целевой функции выступает некоторая оценка качества переходного процесса, получаемого при конкретных значениях параметров регулятора. Метод является итерационным, т.е. в регулятор необходимо последовательно подставлять

все имеющиеся в популяции наборы параметров и, для оценки их оптимальности, проводить опыты на реальном объекте. Или иметь точную модель ОУ, что, как уже упоминалось выше, является достаточно сложной проблемой.

Настройка ПИД-регулятора с помощью нечеткой логики [10, 33, 34, 42, 68, 76, 79] и экспертных систем [21] может осуществляться без использования модели ОУ. В частности, в экспертных системах в базе правил указано на сколько процентов следует изменить амплитудное значение того или иного коэффициента регулятора в зависимости от текущей ситуации, описанной в условии конкретного правила. Проблемой подобных подходов является отсутствие механизма обучения, поскольку первоначально степень нелинейности ОУ достаточно тяжело оценить. В связи с чем в базу правил (в частности, в их следствия) ЭС необходимо будет вносить изменения уже в процессе работы. В случае с использованием нечеткой логики для создания настройщика параметров регулятора его входные и выходные переменные определяют в диапазоне $[-1;1]$ или $[0;1]$. Реальные же измеряемые на ОУ значения параметров затем приводятся в данные диапазоны с помощью коэффициентов нормировки. Также поступают с получаемыми с выходов настройщика значениями K_P , K_I , K_D (их переводят из промежутка $[0;1]$ в диапазон, требуемый для данного ОУ). Данные коэффициенты нормировки также необходимо подстраивать в процессе работы.

Нейронные сети (НС), в свою очередь, обладают нелинейными свойствами и способностью к обучению (в том числе оперативному), что придает адаптивные свойства нейросетевым системам управления. В частности, Сигеру Омату в [58] предложил схему управления на основе автонастройки коэффициентов ПИД-регулятора K_P , K_I , K_D с помощью НС (рис. 1, где e – рассогласование).

Однако реализация идей С. Омату в том виде, в котором они были изложены в его трудах, не всегда позволяет получать заявленные результаты. Это во многом связано с тем, что НС, обучаясь по некоторой обучающей выборке, не обладает в итоге сведениями о специфике того объекта, которым управляет

(наличие ограничений, например, на значения управляющего воздействия; несимметричность ОУ – для печей управление процессом остывания отличается от процессов нагрева и пр.).



Рис. 1. Схема управления с нейросетевым оптимизатором параметров регулятора

Использование схемы С. Омату нашло отражение в трудах многих ученых и разработчиков [3, 6, 28–30, 36, 37, 61, 62, 74]. Однако каждое конкретное применение требовало сложной детальной настройки под конкретную задачу и объект управления (в частности – наличие нейросетевой модели ОУ).

В целом главным преимуществом НС является возможность оперативного дообучения, в то время как специфика работы ОУ может быть задана, например, набором правил, определяющим ситуации, в которых обучение НС целесообразно. Поэтому в рамках данной работы предлагается скомбинировать метод применения НС для коррекции коэффициентов регулятора с базой правил ситуаций, когда необходимо проводить такую настройку. В качестве следствий в правилах выступают скорости обучения нейронов НС [4].

Целью данной работы является разработка методики выбора структуры и параметров нейронной сети, производящей настройку параметров линейного регулятора, по априорным знаниям об ОУ без проведения процедуры идентификации. Кроме того, в рамках данного исследования рассмотренная выше схема была реализована в среде Matlab. Проведен ряд модельных экспериментов, подтверждающих применимость нейросетевого оптимизатора, построенного на основе предложенного подхода, к ОУ с различной динамикой. Данный подход

также был проверен на лабораторных муфельных электронагревательных печах СНОЛ-1,6,2,5.1/11-И4 и СНОЛ 40/1200. Было проведено сравнение работы систем управления с нейросетевым оптимизатором, включающим в себя две нейронных сети, и обычным ПИ-регулятором для различных режимов работы ОУ по загрузке и графику заданий.

2. Постановка задачи

Объектом исследований является схема управления, представленная на рис. 1. ОУ описывается двумя апериодическими звеньями первого порядка со звеном задержки (1). Первое из них является моделью самого ОУ, а второе – моделью средства измерения значений контролируемого параметра для рассматриваемого ОУ:

$$(1) \quad W(s) = \frac{K}{T_1 s + 1} \cdot \frac{1}{T_2 s + 1} \cdot e^{-\tau \cdot s},$$

где $W(s)$ – передаточная функция ОУ, K – коэффициент усиления ОУ; T_1 – постоянная времени ОУ, с; T_2 – постоянная времени измерительного прибора, с; τ – время запаздывания, с.

Таким образом, процессы, происходящие внутри ОУ, могут быть представлены в следующем виде:

$$(2) \quad T_1 T_2 y''(t) + (T_1 + T_2) y'(t) + y(t) = K \cdot u(t - \tau),$$

где u – управляющее воздействие; t – текущий момент времени, с; $y(t)$ – выход ОУ в текущий момент времени.

ПИД-регулятор имеет вид

$$(3) \quad u(t) = K_p e(t) + K_I \int_0^t e(\tau) d\tau + K_D \frac{de(t)}{dt},$$

где $e(t)$ – рассогласование между заданием и выходом ОУ; K_p , K_I , K_D – настраиваемые параметры регулятора. Рассогласование вычисляется как разность сигнала задания (уставки) $r(t)$ и выхода ОУ $y(t)$.

Уравнение (2), используя (3), можно представить в виде

$$(4) \quad T_1 T_2 y''(t) + (T_1 + T_2) y'(t) + y(t) = K \cdot (K_p e(t - \tau) + K_I \int_0^{t-\tau} e(p) dp + K_D \frac{de(t - \tau)}{dt}).$$

Такое уравнение отражает одно конкретное состояние ОУ, поскольку T_1 и K здесь являются константами. Для реального ОУ они являются переменными. В частности, для нагревательных объектов постоянная времени печи может изменяться в зависимости от количества материала, загруженного в эту печь. Коэффициент усиления уменьшается по мере увеличения температуры в печи, а кроме того, зависит от качества энергоносителя, подаваемого на печь.

Задачей контура управления, изображенного на рис. 1, является отслеживание графика уставок, представляющего собой набор ступенчатых изменений задания, таким образом, чтобы, с одной стороны, переходные процессы протекали за минимальное количество времени. Это позволит интенсифицировать производство и сократить потери энергоносителя. С другой стороны, качество получаемых переходных процессов по переуправлению, колебательности, статической ошибке должно находиться в пределах допуска.

Соответственно, для поддержания времени переходного процесса близким к минимальному в условиях изменения параметров T_1 и K необходимо оперативно в процессе работы настраивать коэффициенты регулятора K_p , K_I , K_D .

ОУ уже функционирует под управлением П-, ПИ- или ПИД-регулятора и его коэффициенты изначально подобраны для одного из состояний объекта. Задачей является подстройка (а не полное переопределение) коэффициентов регулятора в соответствии с изменениями режима работы ОУ для поддержания времени переходных процессов для всех режимов близким к минимальному.

Для этого предлагается применить нейросетевой оптимизатор параметров линейных регуляторов. Он не предполагает построение модели ОУ. Вместо этого производится оценка ситуации на ОУ (в частности, текущей скорости переходного

процесса) с помощью базы правил с дальнейшими выводами о необходимости обучения НС.

3. Структура нейросетевого оптимизатора

Предлагаемая функциональная схема нейросетевого оптимизатора приведена на рис. 2. Количество выходов оптимизатора зависит от числа настраиваемых параметров регулятора (например, для ПИД – 3, для ПИ – 2).

Авторами установлено [4], что количество НС в нейросетевом оптимизаторе зависит от того, является ли ОУ симметричным [43] или нет. Например, при работе с нагревательными объектами необходимо вести управление двумя различными по своей природе процессами: нагреванием (когда новая уставка больше предыдущей) и охлаждением (когда новая уставка меньше предыдущей). В соответствии с [43], такой ОУ является несимметричным, т.е. для оптимальности ведения и того, и другого процесса требуется два набора коэффициентов регулятора, а не один, поскольку нагревательная печь неспособна вести принудительное охлаждение внутреннего пространства.



Рис. 2. Разработанная функциональная схема нейросетевого оптимизатора параметров линейных регуляторов

В связи с этим предлагается реализовывать нейросетевой оптимизатор в виде двух идентичных нейронных сетей, каждая из которых отвечает за управление коэффициентами регулятора

для своего типа процесса (для печей – нагрева и остывания). Если ОУ симметричный – достаточно одной НС.

При наличии более одной НС переключение между ними предлагается производить в момент очередной смены задания. При этом определяется тип переходного процесса и производится выбор соответствующей сети. Все НС оптимизатора начинают работу с формирования на своем выходе одинаковых коэффициентов регулятора, используемых на ОУ в текущий момент (до внедрения оптимизатора). Метод достижения этой цели, основанный на экстремальном подходе к обучению НС [48], описан в [5]. Затем производится их оперативное обучение методом обратного распространения ошибки [58].

База правил содержит в себе описание ситуаций, когда необходимо вести оперативное обучение НС, а также скорости обучения для ее отдельных нейронов. Подробно данная база приведена в [4]. Разработанный обобщенный алгоритм работы нейросетевого оптимизатора параметров линейных регуляторов приведен на рис. 3.

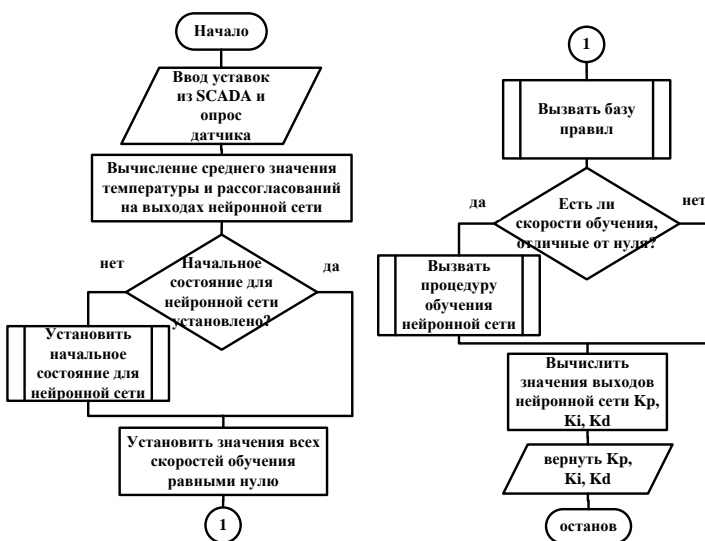


Рис. 3. Разработанный обобщенный алгоритм работы нейросетевого оптимизатора

До начала использования оптимизатора создаются одна или две одинаковых необученных НС. Входные сигналы для НС – задание $r(t)$, выход ОУ $y(t)$, выход линейного регулятора $u(t)$. Выход НС – значения настраиваемых параметров линейного регулятора. Фактически НС представляет собой функцию нескольких переменных, отражающую связь K_P, K_I, K_D с указанными входными параметрами, в частности, $K_P = f^1(r^{-1}, r^{-2}, \dots, r^{-n}, y^{-1}, y^{-2}, \dots, y^{-m}, u^{-1}, u^{-2}, \dots, u^{-k})$. НС аппроксимирует подобные зависимости. Принципы выбора структуры такой НС приведены ниже.

4. Метод выбора структуры нейронной сети

4.1. ОПРЕДЕЛЕНИЕ ОБЩЕЙ СТРУКТУРЫ СЕТИ

Результатом серии работ [2, 7, 52] А.Н. Колмогорова и В.И. Арнольда стало доказательство теорем о возможности представления непрерывных функций нескольких переменных в виде суперпозиций непрерывных функций одной переменной.

В работе [44] Р. Хехт-Нильсен предложил применить теорему А.Н. Колмогорова для обоснования возможности нейронных сетей аппроксимировать непрерывные функции многих переменных со сколь угодно малой погрешностью.

Строгое математическое доказательство универсальности аппроксимационных свойств двухслойной нейронной сети (входной слой, один скрытый слой, выходной слой) с сигмоидальной функцией (в частности, гиперболический тангенс) активации в скрытом слое и линейной в выходном было дано независимо тремя авторами в работах [32, 39, 45].

В работе [41] доказана универсальная аппроксимационная теорема, обобщающая как результаты А.Н. Колмогорова и Р. Хехт-Нильсена и работ [32, 39, 45], так и аппроксимационную теорему Стоуна–Вейерштрасса [65]. С точки зрения нейронных сетей ее смысл сводится к тому, что для сколь угодно точной аппроксимации любой функции многих переменных достаточно двухслойной структуры НС при условии, что в скрытом слое использована нелинейная функция активации.

Таким образом, основываясь на всех приведенных выше результатах, для решения поставленной задачи будет достаточно двухслойной НС: входной слой – скрытый слой – выходной слой. В скрытом слое должна быть сигмоидальная функция активации, а в выходном – линейная.

4.2. ОПРЕДЕЛЕНИЕ КОЛИЧЕСТВА НЕЙРОНОВ ВО ВХОДНОМ И ВЫХОДНОМ СЛОЯХ НС

Рассмотрим проблему определения количества нейронов во входном слое НС для решения задачи настройки параметров П-, ПИ-, ПИД-регуляторов.

В работах [13, 26] эта проблема рассматривается для задачи построения нейрорегулятора. Несмотря на то, что в данном случае нейронная сеть не заменяет регулятор, а настраивает его, некоторые положения указанных исследований могут быть применены. И в частности тот факт, что нейронная сеть на своих входах должна иметь не меньше информации, чем имеет регулятор. Определим, что включает в себя эта информация.

Рассмотрим ПИД-регулятор. Зависимость выхода регулятора от входа может быть описана в виде (3). В операторной форме уравнение (3) примет вид (5) ($U(s)$ и $Y(s)$ – преобразования Лапласа для $u(t)$ и $y(t)$ соответственно):

$$(5) \quad U(s) = \left(K_p + \frac{K_I}{s} + K_D s \right) Y(s).$$

Соответственно, передаточная функция регулятора $W_c(s)$ примет вид

$$(6) \quad W_c(s) = U(s)/Y(s) = K_p + \frac{K_I}{s} + K_D s.$$

Нейронная сеть, используемая для решения указанной задачи, является дискретной и вызывается раз в Δt секунд. То есть хотя сам регулятор и является непрерывным (условно, поскольку реализован средствами ЭВМ и промышленного контроллера, однако их такт работы намного меньше Δt , поэтому можно заключить, что Δt на его работу не влияет), его работа оценивается нейронной сетью дискретно, поэтому с целью определения состава нейронов входного слоя такой сети переведем данный

закон управления (6) в дискретный вид (7), воспользовавшись заменой $s = (1 - z^{-1}) / \Delta t$:

$$\begin{aligned}
 W_c(z) &= K_P + \frac{K_I \Delta t}{1 - z^{-1}} + K_D \frac{1 - z^{-1}}{\Delta t} = \frac{1}{1 - z^{-1}} (K_P (1 - z^{-1}) + \\
 (7) \quad &+ K_I \Delta t + \frac{K_D}{\Delta t} (1 - z^{-1})^2) = \frac{1}{1 - z^{-1}} \left(\frac{K_D}{\Delta t} z^{-2} - \right. \\
 &\left. - (2 \frac{K_D}{\Delta t} + K_P) z^{-1} + (K_P + K_I \Delta t + \frac{K_D}{\Delta t}) \right).
 \end{aligned}$$

Обозначим $a_1 = (K_P + K_I \Delta t + K_D / \Delta t)$, $a_2 = - (2 K_D / \Delta t + K_P)$, $a_3 = K_D / \Delta t$. Таким образом, разностное уравнение, являющееся аналогом (5), для k -го шага управления (момента времени t_k) имеет вид

$$(8) \quad u(t_k) = a_1 e(t_k) + a_2 e(t_k - \Delta t) + a_3 e(t_k - 2\Delta t) + u(t_k - \Delta t).$$

То есть ПИД-регулятор фактически имеет сведения о рассогласовании в текущий момент, такт назад, два такта назад и о сигнале управления такт назад. Авторы [13, 26] делают вывод о том, что нейронная сеть, заменяющая ПИД-регулятор, должна иметь не меньше информации, чем имеет регулятор, поэтому НС должна иметь 4 входа.

В данном случае вывод имеет сходный вид: настройщик (нейросетевой оптимизатор) ПИД-регулятора не может иметь информации меньше, чем имеет сам регулятор. Но существует одно важное различие. Настройщик должен знать не просто рассогласование в текущий момент, но и текущее значение уставки, поскольку при разных уставках при одинаковом рассогласовании могут быть необходимы различные значения параметров регулятора. В уравнении (8) $e(t)$ есть разница уставки $r(t)$ и выхода ОУ $y(t)$. При этом следует учитывать тот факт, что при построении нейросетевого оптимизатора параметров линейных регуляторов рассматриваются системы управления со ступенчатой сменой уставок (см. п. 2), а настройка ведется только в течение переходного процесса. В таких условиях уставку для конкретного переходного процесса следует считать константой, не зависящей от времени. Тогда уравнение (8) примет вид

$$(9) \quad u(t_k) = (a_1 + a_2 + a_3)r - a_1 y(t_k) - a_2 y(t_k - \Delta t) - a_3 y(t - 2\Delta t) + u(t_k - \Delta t).$$

Таким образом, нейронная сеть для нейросетевого оптимизатора параметров ПИД-регулятора должна содержать пять входов: задание, выход ОУ в текущий момент, такт назад, два такта назад и значение управляющего воздействия в предыдущий момент времени.

Для других законов регулирования, основанных на П-, И-, Д-составляющих, структура входного слоя НС может быть получена таким же образом. Однако НС с пятью входами в полной мере обеспечит настройку как П-, так и ПИ- или ПИД-регулятора.

Количество нейронов в выходном слое определяется количеством настраиваемых параметров регулятора. НС для случая с ПИД-регулятором должна иметь 3 выхода, каждый из которых отвечает за вычисление соответствующего параметра регулятора K_P , K_I , K_D . В выходном слое, в соответствии с [32, 39, 45], должна использоваться линейная функция активации.

4.3. ФУНКЦИЯ АКТИВАЦИИ СКРЫТОГО СЛОЯ НС

В соответствии с рассмотренными ранее теоремами в качестве функции активации скрытого слоя НС должна быть использована нелинейная функция сигмоидального вида. Наиболее часто применимыми функциями являются непосредственно сигмоидальная функция (рис. 4а) $f(s) = 1/(1 + e^{-\alpha s})$, $\alpha = \text{const}$, и гиперболический тангенс $f(s) = (e^{\alpha s} - e^{-\alpha s})/(e^{\alpha s} + e^{-\alpha s})$ (рис. 4б).

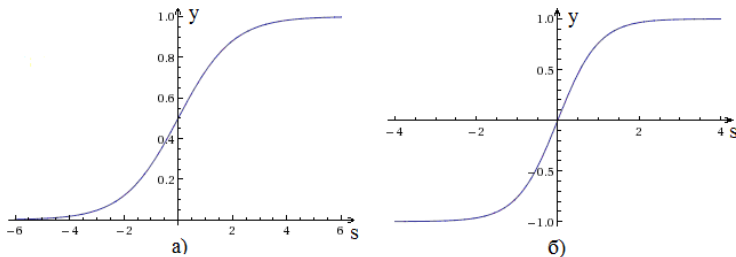


Рис. 4. Обычная сигмоидальная функция (а) и гиперболический тангенс (б)

Они имеют сходные преимущества, в частности, у обеих функций производная может быть легко выражена через саму функцию, что позволяет существенно сократить вычислительную сложность метода обучения (обратного распространения ошибки). Основное различие между ними заключается в том, что область определения для обычной сигмоидальной функции – $[0;1]$, а для гиперболического тангенса – $[-1;1]$.

Для того чтобы осуществить выбор между этими двумя вариантами, необходимо определить допустимы (необходимы) ли отрицательные значения на выходе скрытого слоя НС. Для ответа на этот вопрос следует детальнее рассмотреть принцип работы нейросетевого оптимизатора [4]. Его база правил определяет, когда учить НС, с какой скоростью учить нейроны выходного слоя (причем для отдельных нейронов выходного слоя эта скорость может отличаться) и, самое главное, в каком направлении следует изменять текущее значение параметров регулятора K_P , K_I , K_D (уменьшать или увеличивать). Направление (т.е. знак коррекции весовых коэффициентов между скрытым и выходным слоями НС $\Delta\omega_{kj}^{(2)}(t)$), в соответствии с математической моделью метода обратного распространения ошибки (10) [58], должно определяться знаком скорости обучения нейрона выходного слоя $\eta_k^{(2)}$:

$$\Delta\omega_{kj}^{(2)}(t) = -\eta_k^{(2)} \delta_k^{(2)} O_j^{(1)},$$

$$(10) \quad \omega_{kj}^{(2)}(t+1) = \omega_{kj}^{(2)}(t) + \Delta\omega_{kj}^{(2)}(t),$$

$$\delta_k^{(2)} = e_k;$$

где $\eta_k^{(2)}$ – это скорость обучения для k -го нейрона выходного слоя; $\delta_k^{(2)}$ – это совокупная ошибка k -го нейрона выходного слоя; e_k – это ошибка k -го нейрона выходного слоя (для каждого из каналов K_P , K_I , K_D используется своя формула для расчета данного значения – см. п. 4.5); $O_j^{(1)}$ – это выходной сигнал j -го нейрона скрытого слоя; $\omega_{kj}^{(2)}$ – весовой коэффициент между k -м нейроном выходного слоя и j -м нейроном скрытого слоя.

А для того чтобы, меняя знак скорости обучения $\eta_k^{(2)}$, возможно было менять направление изменения выхода нейрона выходного слоя, все остальные множители в формуле коррек-

ции такого весового коэффициента ($\delta_k^{(2)}$ и $O_j^{(1)}$) данной математической модели должны иметь постоянные знаки. Для совокупной ошибки нейрона $\delta_k^{(2)}$ это невозможно, поскольку она меняет знак в зависимости от знака рассогласования e_k , однако все эти ситуации учтены в правилах базы правил нейросетевого оптимизатора. А вот выход нейронов скрытого слоя $O_j^{(1)}$ должен иметь постоянный знак, например, быть неотрицательным. Поэтому в данном случае целесообразно выбрать сигмоидальную функцию активации.

4.4. ОПРЕДЕЛЕНИЕ КОЛИЧЕСТВА НЕЙРОНОВ В СКРЫТОМ СЛОЕ НС

В соответствии с [66] первоначально необходимо определить, к какому классу относится создаваемая НС: сужающаяся (в скрытом слое нейронов меньше, чем во входном) или расширяющаяся сеть (в скрытом слое нейронов больше, чем во входном). Чем выше требования к различению близких входных векторов, тем более расходящейся должна быть сеть [66]. Основываясь на данном утверждении, можно сделать вывод о том, что в случае с нейросетевым оптимизатором параметров линейных регуляторов сеть должна быть расширяющейся, поскольку похожие входные вектора могут требовать различных значений коэффициентов регулятора (ввиду нелинейности ОУ).

В целом обзор по методикам выбора количества нейронов в скрытом слое приведен в [64]. Данные методы можно разделить на две большие группы: конструктивные [25, 35, 38, 60, 78] и аналитические [22, 44, 46, 47, 57, 67, 69, 72, 73].

Конструктивные методы предполагают добавление (удаление) нейронов в скрытом слое с отслеживанием изменения критерия качества обучения: ошибки обучения, ошибки обобщения и пр. Все конструктивные методы применимы для автономного обучения НС, когда есть готовая обучающая выборка и достаточный объем времени для проведения описанных итеративных процедур. В случае с нейросетевым оптимизатором линейных регуляторов автономная обучающая выборка отсутствует, а обучение необходимо вести в оперативном режиме в

реальном масштабе времени. Поэтому применение указанных методов в данном случае является затруднительным.

В свою очередь, аналитические методы предполагают наличие некоторой математической формулы, позволяющей оценить количество нейронов в скрытом слое НС в зависимости от параметров самой НС и характеристик предметной области, с которой НС предстоит работать.

В частности, в работе [44] Р. Хехт-Нильсен, опираясь на теорему Колмогорова–Арнольда, показал, что в скрытом слое достаточно наличия $(2N + 1)$ нейронов, где N – это число входов НС. Однако уже в работе [54] было обосновано, что число нейронов скрытого слоя должно удовлетворять следующему неравенству: $N_{hid} \geq 2N + 1$. Данный подход может эффективно применяться в качестве нижней оценки количества нейронов. Тогда в скрытом слое НС для ПИД-регулятора должно быть не меньше **11 нейронов**. Подобная оценка может быть легко проведена для любого линейного регулятора, содержащего П-, И-, Д- каналы.

В [46, 47, 67] доказано, что НС прямого распространения с одним скрытым слоем с N нейронами и практически любой нелинейной функцией активации (дифференцируемой и недифференцируемой, непрерывной и с разрывами) в скрытом слое может запомнить и обобщить N различных примеров.

Исходя из работ [48, 49], количество нейронов скрытого слоя и объем обучающей выборки должны совпадать для достижения близкой к нулю ошибки. Для нейросетевого оптимизатора обучение НС ведется в течение конкретного переходного процесса, для которого подбираются параметры регулятора. Тогда можно сделать вывод о том, что количество точек конкретного переходного процесса, которые необходимо предъявить нейронной сети для анализа, должно совпадать с количеством нейронов в скрытом слое. Исходя из оценок Р. Хехт-Нильсена, приведенных выше, получается, что таких точек должно быть не менее 11 для ПИД-регулятора.

Но необходимо учесть еще один факт. Как уже упоминалось ранее, на производстве для борьбы с шумом измеряемые величины – характеристики ОУ или процесса подвергаются

усреднению, причем количество усредняемых точек может быть различным. Например, если усреднение ведется по трем точкам, то для получения 11 усредненных результатов о конкретном переходном процессе после момента смены уставки в реальности потребуется $3 + 1 + 1 + 10 = 15$ измерений. Первые три реальные точки позволят получить первый усредненный выход ОУ, а для случая с ПИД-регулятором необходимо еще иметь выход ОУ такт и два назад. Поэтому для их вычисления потребуется еще $1 + 1 = 2$ измерения. Таким образом будет получен первый входной вектор для НС. А кроме него потребуется еще 10 таких векторов. То есть в данном случае количество нейронов в скрытом слое также должно быть равно 15.

Обобщая полученные результаты, для нейросетевого оптимизатора линейных регуляторов количество нейронов в скрытом слое предлагается вычислять следующим образом:

$$(11) N_{hid} = (2N + 1) + N_{av} + N_{delay} - 1,$$

где N_{hid} – количество нейронов в скрытом слое НС; N – число входов НС; N_{av} – число точек, по которым ведется усреднение данных, получаемых с ОУ (затем эти усредненные данные нормируются и подаются на входы НС); N_{delay} – число входов НС, являющихся задержанными сигналами по выходу ОУ.

И это же число N_{hid} , основываясь на результатах работ [48, 49], определяет количество вызовов НС за время одного переходного процесса. Данный факт будет необходим для п. 4.6.

В экспериментах, проведенных в рамках данной работы, усреднение велось именно по трем точкам. Поэтому структура НС для настройки ПИД-регулятора имеет вид: 5-15-3.

4.5. ОПИСАНИЕ ВЫБРАННОЙ СТРУКТУРЫ НС И МЕТОДА ЕЕ ОПЕРАТИВНОГО ОБУЧЕНИЯ

Выбранная структура нейронной сети приведена на рис. 5. Математическая модель сети представлена в виде (12). Во входном слое расположено 5 нейронов: текущее задание (x_1); задержанный на 1 секунду (x_2), на Δt секунд (x_3) и на $2\Delta t$ секунд (x_4) сигнал по выходу ОУ; задержанный на 1 секунду сигнал управления, формируемый ПИД-регулятором (x_5). Входные сигналы

нормируются в интервал $[0;1]$. Обращение к НС также происходит раз в Δt секунд.

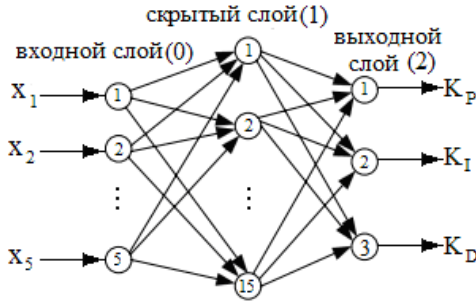


Рис. 5. Структура нейронной сети, применяемой в нейросетевом оптимизаторе параметров ПИД-регулятора

$$s_j^{(1)} = \sum_{i=1}^5 \omega_{ji}^{(1)} \cdot x_i + b_j^{(1)},$$

$$O_j^{(1)} = f^{(1)}(s_j^{(1)}) \quad (j = \overline{1, 15}),$$

$$s_k^{(2)} = \sum_{j=1}^{15} \omega_{kj}^{(2)} \cdot O_j^{(1)} + b_k^{(2)},$$

$$O_k^{(2)} = f^{(2)}(s_k^{(2)}) \quad (k = \overline{1, 3});$$

где $\omega_{ji}^{(1)}$ – весовой коэффициент связи между j -м нейроном скрытого слоя и i -м нейроном входного слоя; $\omega_{kj}^{(2)}$ – весовой коэффициент связи между k -м нейроном выходного слоя и j -м нейроном скрытого слоя; x_i – i -й входной сигнал НС; $b_j^{(1)}$ – линейное смещение j -го нейрона скрытого слоя; $b_k^{(2)}$ – линейное смещение k -го нейрона выходного слоя; $O_j^{(1)}$ – выходной сигнал j -го нейрона скрытого слоя; $O_k^{(2)}$ – выходной сигнал k -го нейрона выходного слоя; $s_j^{(1)}$ – взвешенная сумма для j -го нейрона скрытого слоя; $s_k^{(2)}$ – взвешенная сумма для k -го нейрона выходного слоя; $f^{(q)}$ – функция активации слоя номер q . В частности, $f^{(1)}$ – гиперболический тангенс; $f^{(2)}$ – линейная функция активации; $O_1^{(2)}$ – это K_P , $O_2^{(2)}$ – это K_I , $O_3^{(2)}$ – это K_D .

Все НС нейросетевого оптимизатора предлагается оперативно обучать с помощью алгоритма обратного распространения ошибки. Математическая модель метода в адаптированном для рассматриваемой НС виде представлена в виде (13)–(18). Общей целью обучения является минимизация рассогласования между выходом ОУ $y(t)$ и заданием $r(t)$:

$$(13) \quad E(t) = \frac{1}{2} (r(t) - y(t))^2 \rightarrow \min ,$$

$$(14) \quad \begin{aligned} e_2(t) &= r(t) - y(t), \quad e_1(t) = e_2(t) - e_2(t-1), \\ e_3 &= e_2(t) - 2e_2(t-1) + e_2(t-2), \end{aligned}$$

$$(15) \quad \delta_k^{(2)} = e_k \frac{dO_k^{(2)}}{ds_k^{(2)}}, \quad k = \overline{1, 3},$$

$$(16) \quad \delta_j^{(1)} = \sum_{k=1}^3 \delta_k^{(2)} \omega_{kj}^{(2)} \frac{dO_j^{(1)}}{ds_j^{(1)}}, \quad j = \overline{1, 15},$$

$$(17) \quad \begin{aligned} \Delta \omega_{kj}^{(2)}(t) &= \eta_k^{(2)} \delta_k^{(2)} O_j^{(1)} + \alpha \Delta \omega_{kj}^{(2)}(t-1) + \beta \Delta \omega_{kj}^{(2)}(t-2), \\ \Delta b_k^{(2)}(t) &= \eta_k^{(2)} \delta_k^{(2)} + \alpha \Delta b_k^{(2)}(t-1) + \beta \Delta b_k^{(2)}(t-2), \\ \Delta \omega_{ji}^{(1)}(t) &= \eta_j^{(1)} \delta_j^{(1)} O_i^{(0)} + \alpha \Delta \omega_{ji}^{(1)}(t-1) + \beta \Delta \omega_{ji}^{(1)}(t-2), \\ \Delta b_j^{(1)}(t) &= \eta_j^{(1)} \delta_j^{(1)} + \alpha \Delta b_j^{(1)}(t-1) + \beta \Delta b_j^{(1)}(t-2), \\ \omega_{kj}^{(2)}(t+1) &= \omega_{kj}^{(2)}(t) + \Delta \omega_{kj}^{(2)}(t), \\ b_k^{(2)}(t+1) &= b_k^{(2)}(t) + \Delta b_k^{(2)}(t), \\ \omega_{ji}^{(1)}(t+1) &= \omega_{ji}^{(1)}(t) + \Delta \omega_{ji}^{(1)}(t), \\ b_j^{(1)}(t+1) &= b_j^{(1)}(t) + \Delta b_j^{(1)}(t); \end{aligned}$$

где $E(t)$ – целевая функция процесса оперативного обучения; $r(t)$ – это задание по температуре; $y(t)$ – выход ОУ; $\eta^{(1)}$ – скорость обучения для нейронов скрытого слоя; $\eta_k^{(2)}$ – скорость обучения для k -го нейрона выходного слоя; α и β – коэффициенты инерции, применяемые для ускорения сходимости; $\delta_j^{(1)}$ – совокупная ошибка j -го нейрона скрытого слоя; $\delta_k^{(2)}$ – совокупная ошибка k -го нейрона выходного слоя;

e_k – ошибка k -го нейрона выходного слоя; $dO_j^{(q)}/ds_j^{(q)}$ – производная функции активации слоя q .

В соответствии с [58] параметр α был принят равным 0,8, а β – равным $(-0,15)$. Скорость обучения для всех нейронов скрытого слоя $\eta^{(1)}$ является одинаковой и не меняется в течение работы системы. Численное значение данной величины также заимствовано из работы [58] и составляет 10^{-4} . Оптимальность выбранных значений α , β и $\eta^{(1)}$ была проверена экспериментально. Их уменьшение ведет к увеличению количества переходных процессов, требуемых для перенастройки параметров регулятора в результате смены состояния объекта управления. В случае увеличения значений указанных параметров скорость изменения выходных сигналов сети в процессе обучения возрастает, что в некоторых случаях приводит к колебаниям значений K_P , K_I около оптимального значения.

В отличие от [58], в данной работе предлагается сделать значения $\eta_k^{(2)}$ различными для нейронов выходного слоя НС и изменять их в течение работы системы. Пусть $\eta_1^{(2)}$ – скорость коррекции весовых коэффициентов и смещения, отвечающих за формирование выхода НС $O_1^{(2)}$, т.е. K_P . Соответственно, $\eta_2^{(2)}$ отвечает за K_I и $\eta_3^{(2)}$ – за K_D . Необходимость такого подхода вызвана, во-первых, тем фактом, что порядок значений коэффициентов регулятора обычно различается (например, $K_P = 2 \cdot 10^{-1}$, а $K_I = 1,5 \cdot 10^{-4}$), поэтому и скорость их настройки должна быть пропорциональна этому значению. Во-вторых, нет необходимости изменять коэффициенты регулятора постоянно, а в некоторые моменты коррекции требует только один из них.

Поэтому в начале каждой итерации работы нейросетевого оптимизатора все три указанные скорости приравниваются к нулю, а затем происходит вызов разработанной базы правил. Условия правил описывают ситуации, когда необходимо использовать и обучать НС, а следствия – необходимые значения $\eta_1^{(2)}$, $\eta_2^{(2)}$, $\eta_3^{(2)}$. Если какое-либо из правил срабатывает и при этом сигнал выхода ПИД-регулятора находится в допустимых пределах, то происходит обучение НС.

Важным вопросом также остается выбор значения Δt для каждого конкретного ОУ, поскольку данный параметр во многом определяет точность работы НС.

4.6. ОПРЕДЕЛЕНИЕ ЗНАЧЕНИЙ ЗАДЕРЖЕК ВХОДНЫХ СИГНАЛОВ НС НЕЙРОСЕТЕВОГО ОПТИМИЗАТОРА

Значение Δt – это и дискретность вызова НС, и время, на которое отстоят друга от друга задержанные сигналы по выходу ОУ, являющиеся входами НС.

Одним из возможных путей решения проблемы выбора Δt является включение этого параметра в алгоритм обучения (например, обратного распространения ошибки) в качестве настраиваемого параметра [51]. Однако это, во-первых, усложняет алгоритм обучения, требуя больше времени и ресурсов. А во-вторых, такой подход применим при автономном обучении, когда есть время и возможность без последствий перебирать значения Δt . В работах [40, 77] предложен итеративный метод выбора структуры НС и времени задержки входных сигналов при построении нейросетевой модели технологического процесса с одним входом – одним выходом и множеством входов – множеством выходов. Данный подход имеет тот же недостаток, что и описанный ранее – он применим только для автономного обучения.

Для выбора значения Δt в задаче построения НС для нейросетевого оптимизатора параметров линейных регуляторов предлагается воспользоваться результатами, описанными в п. 4.4 данной работы. Количество нейронов в скрытом слое, метод определения которого был предложен в п. 4.4, совпадает с числом вызовов НС в течение конкретного переходного процесса. Нейросетевой оптимизатор устанавливается уже в работающую систему управления, где известны текущие значения параметров регулятора (они позволяют системе работать, но не являются оптимальными) и время переходных процессов под управлением данного регулятора для данных значений его параметров. Тогда Δt можно определить, выбрав самый короткий в текущей картине заданий переходный процесс и разделив его на число нейронов в скрытом слое НС. Этого достаточно

для того, чтобы в течение конкретного переходного процесса НС была вызвана достаточное число раз.

В дальнейшем планируется ввести в оптимизатор функцию расчета времени Δt , которая будет автоматически перерасчитывать это значение (в сторону уменьшения), если было получено недостаточное количество измерений выхода ОУ.

Для проверки предложенного подхода была проведена серия модельных экспериментов по определению оптимального значения Δt для различных сочетаний значений параметров ОУ, описываемого с помощью (1). Результаты приведены в п. 5.

5. Результаты моделирования работы нейросетевого оптимизатора для различных значений параметров ОУ с целью выбора Δt

База правил, приведенная в [4], разработана для настройки П- и И-каналов регулятора. Поэтому дальнейшие эксперименты проведены для ПИ-регулятора. При этом в экспериментах использовалась НС следующей структуры: 5-15-3. Однако так как производилась настройка ПИ-регулятора, третий выход НС всегда был равен нулю и обучение соответствующего ему нейрона не производилось ($\eta_3^{(2)} = 0$).

В пакете Matlab была реализована система управления моделью ОУ (рис. 6), описываемой (1).

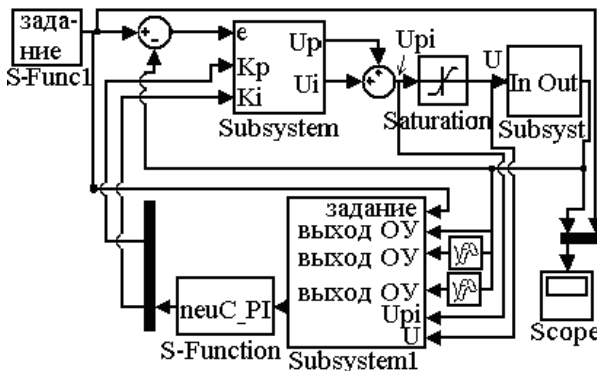


Рис. 6. Система управления с нейросетевым оптимизатором

Контур управления представлен блоком *Subsystem* с ПИ-регулятором с нейросетевым оптимизатором (*S-function neuC_PI*). U_i , U_p , U_{pi} – сигналы управления на выходе i -, p -каналов и всего ПИ-регулятора соответственно, e – текущее рассогласование. Блок *Saturation* использован для ограничения получаемого с ПИ-регулятора сигнала управления U до диапазона $[0;100]$ для придания ОУ, описываемого выражением (1), несимметричных свойств.

Для каждого опыта было использовано по два набора параметров модели (сочетания), между которыми осуществлялось подобное переключение: первое состояние – 12 переходов с параметрами $(T_1, \tau) \rightarrow$ второе состояние – 12 переходов с параметрами $(2 \cdot T_1, 2 \cdot \tau) \rightarrow$ первое состояние – 12 переходов с параметрами (T_1, τ) . Значение T_2 всегда составляло 5% от T_1 . Начальные значения коэффициентов ПИ-регулятора для каждого сочетания были подобраны для первого состояния ОУ. Задачей оптимизатора было сначала подобрать новые коэффициенты для измененного состояния объекта, а затем вернуть исходные параметры при возврате к исходному состоянию ОУ. Для смены уставки выход ОУ должен был удерживаться в 5% окрестности значения разности текущего задания и предшествующего в течение 1800 секунд. Для каждого из сочетаний параметров ОУ также был проведен опыт с обычным ПИ-регулятором, который использовал значения коэффициентов, подобранные для первого состояния ОУ, в течение всего эксперимента.

Всего было проведено 17280 экспериментов: для каждого из трех значений коэффициента усиления ОУ $K = 10, 20$ и 30 было использовано по 144 различных сочетания значений остальных параметров модели ОУ ($T_1 \in [10 \text{ с}; 15000 \text{ с}]$, $\tau \in [1 \text{ с}; 450 \text{ с}]$). Для каждого такого сочетания было проведено по 40 опытов со значениями Δt от 1 секунды до 40 секунд. В каждом таком наборе из 40 опытов выбирался лучший. Для этого был разработан критерий оптимальности в виде суммы следующих значений, нормированных в промежуток $[0;1]$ относительно текущего набора из 40 опытов: время текущего опыта, среднее перерегулирование за весь опыт и сумма квадратов отклонений выхода ОУ от задания за все время опыта. Выби-

рался опыт с минимальным значением критерия среди имеющихся. При этом использовались следующие ограничения: завершены все 36 переходных процессов, среднее перерегулирование за все время опыта не могло быть более 5% от разницы предшествующего и текущего заданий, а общее время опыта должно было быть меньше, чем время, показанное системой с обычным ПИ-регулятором в таких же условиях для такой же модели ОУ. Значение Δt для данного опыта выбиралось как лучшее для данного сочетания параметров ОУ.

В дальнейшем апробация разработанного метода определения структуры и параметров НС произведена на примере муфельных печей, поэтому для моделирования для всех моделей ОУ был выбран следующий график изменения заданий: $590^{\circ}\text{C} \rightarrow 640^{\circ}\text{C} \rightarrow 505^{\circ}\text{C} \rightarrow 590^{\circ}\text{C}$, являющийся уменьшенной в два раза по амплитуде копией графика для печи нагрева литой заготовки перед прокатом.

По итогам моделирования для каждого конкретного набора параметров модели ОУ было определено оптимальное значение параметра Δt . Полученные результаты представлены в виде поверхностей для $K = 10$, $K = 20$ и $K = 30$, отражающих зависимость оптимального значения Δt от T_1 и τ (см. рис. 7–9). Кроме того, по полученным данным аппроксимирован аналитический вид зависимости (19) Δt от K , T_1 и τ .

Для моделей ОУ, в которых $\tau > 0,1T_1$, ни одно из значений Δt не удовлетворяло предъявленным требованиям. Таким образом, для применения нейросетевого оптимизатора необходимо, чтобы $\tau \leq 0,1T_1$.

Экономия по времени на выполнение графика заданий относительно обычного ПИ-регулятора для предлагаемой системы по всем опытам составляло от 3% до 30%, по расходу энергоносителя – от 3% до 31%.

В целом проведенные исследования позволяют сделать вывод о том, что предлагаемый метод настройки параметров регулятора с помощью нейросетевого оптимизатора применим к ОУ с различной динамикой, причем без необходимости изменения структуры НС или базы правил. Однако при этом должно быть выбрано адекватное значение Δt .

$$\begin{aligned}
 \Delta t = & 4.5 - 4.5 \cdot 10^{-12} K^8 + 1.91 \cdot 10^{-23} K^3 T_1^3 \tau^3 + 1.32 \cdot 10^{-15} K^3 T_1^3 - \\
 & - 5.29 \cdot 10^{-12} K^3 T_1^2 + 1.44 \cdot 10^{-8} K^3 T_1 + 1.27 \cdot 10^{-11} K^3 \tau^3 - \\
 & - 8.03 \cdot 10^{-9} K^3 \tau^2 + 3.38 \cdot 10^{-6} K^3 \tau - 5.39 \cdot 10^{-15} K^2 T_1^2 \tau^2 - \\
 & - 3.84 \cdot 10^{-10} K^2 T_1^2 + 1.06 \cdot 10^{-5} T_1^2 / K^2 - 0.12 \cdot T_1 / K^2 - \\
 (19) \quad & - 1.78 \cdot 10^{-7} K^2 \tau^2 - 0.02 \tau^2 / K^2 + 10.45 \tau / K^2 - 320.6 / K^2 - \\
 & - 2.25 \cdot 10^{-6} T_1^2 / K + 2.92 \cdot 10^{-7} K T_1 \tau - 1.39 \cdot 10^{-4} T_1 \tau / K + \\
 & + 2.7 \cdot 10^{-2} T_1 / K - 1.26 \cdot 10^{-3} \tau^2 / K + 0.46 \tau / K + \\
 & + 18.28 / K + 1.84 \cdot 10^{-31} T_1^8 - 8.99 \cdot 10^{-12} T_1^2 \tau^2 + \\
 & + 1.56 \cdot 10^{-9} T_1^2 \tau - 3.49 \cdot 10^{-7} T_1^2 + 1.27 \cdot 10^{-7} T_1 \tau^2 - 1.43 \cdot 10^{-5} T_1 \tau + \\
 & + 4.1 \cdot 10^{-3} T_1 - 1.15 \cdot 10^{-20} \tau^8 - 6.57 \cdot 10^{-5} \tau^2 + 2.28 \cdot 10^{-3} \tau.
 \end{aligned}$$

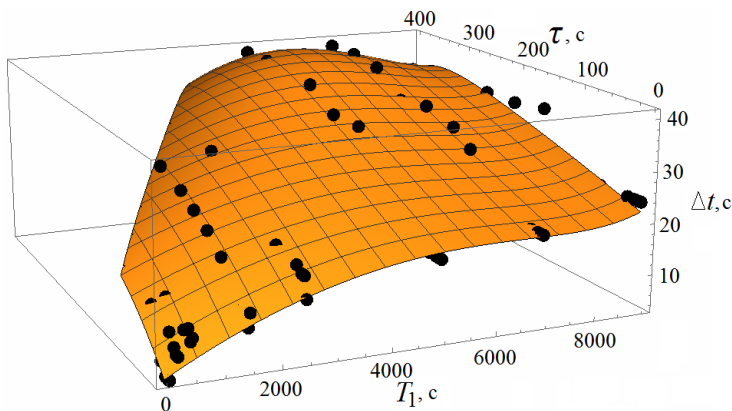


Рис. 7. Зависимость Δt от T_1 и τ при $K = 10$

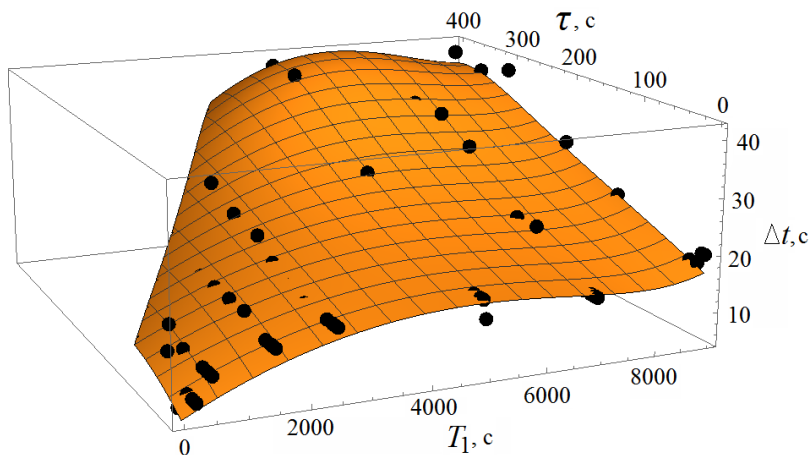


Рис. 8. Зависимость Δt от T_1 и τ для $K = 20$

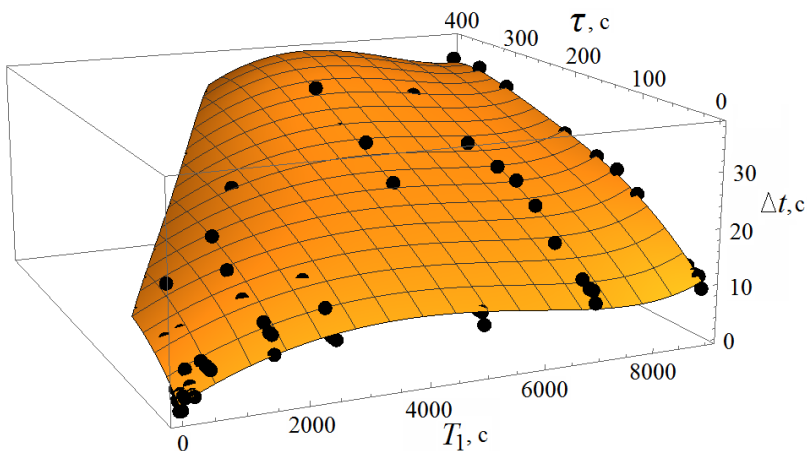


Рис. 9. Зависимость Δt от T_1 и τ для $K = 30$

Для этого может быть использована полученная зависимость (19), но лишь при наличии модели ОУ. Однако в случае

внедрения оптимизатора на реальный ОУ его модель будет неизвестна.

В этом случае для вычисления Δt необходимо будет воспользоваться методом, предложенным в п. 4.6. Проверим адекватность данного метода на основе имеющихся экспериментальных данных. Для этого необходимо показать, что в течение любого переходного процесса нейросетевой оптимизатор в среднем анализирует одинаковое число точек этого процесса (вызывается одинаковым число раз). Причем это число должно совпадать с количеством нейронов в скрытом слое.

При решении данной задачи для каждой из моделей ОУ, используя выбранное оптимальное значение Δt , оценивалось усредненное по всем переходным процессам значение времени T_{III} , в течение которого рассогласование между выходом ОУ и заданием находилось в пределах от 10% до 80% разницы между текущим заданием и предыдущим. Этот промежуток времени важен с точки зрения правил, определяющих моменты обучения нейронной сети. Если НС не будет вызвана в течение этого промежутка времени, то нейросетевой оптимизатор не сможет настроить коэффициенты регулятора, а вызывается такая сеть раз в Δt секунд. Имея значение T_{III} и разделив его на значение Δt для текущей модели ОУ, можно получить, сколько в среднем раз за время переходного процесса был вызван нейросетевой оптимизатор: $N = T_{III} / \Delta t$.

Вычислив значение N для всех опытов, для которых удалось определить оптимальное значение Δt , было замечено, что диапазон значений N составляет [9; 24]. Было решено провести статистический анализ данных результатов. В итоге было получено распределение, приведенное на рис. 10.



Рис. 10. Распределение появления значений N

Согласно критерию Пирсона, данное распределение является нормальным. Математическое ожидание (МО) величины N составляет 15,94, СКО – 3,48. Это дает основание полагать, что для всех сходных экспериментов значение N будет подчиняться данному распределению. Полученное значение математического ожидания величины N практически совпадает с количеством нейронов в скрытом слое НС, что подтверждает адекватность метода вычисления Δt , предложенного в п. 4.6.

По результатам моделирования были выявлены следующие ограничения на применение нейросетевого оптимизатора: 1) задание должно изменяться ступенчато; 2) соотношение транспортного запаздывания к постоянной времени ОУ должно быть менее 0,1 (аналогичное ограничение накладывается, например, на применение адаптивного регулятора фирмы Siemens [59]); 3) минимальная величина ступенчатого изменения задания, для которой нейросетевой оптимизатор сохраняет работоспособность, прямо пропорционально зависит от амплитуды помехи измерения выходной величины ОУ.

На основе рассмотренных выше подходов предлагается общий алгоритм построения и инициализации НС нейросетевого оптимизатора параметров линейных регуляторов.

6. Общий алгоритм выбора структуры НС нейросетевого оптимизатора и ее инициализации

Предлагается следующий обобщенный алгоритм построения и инициализации НС для нейросетевого оптимизатора параметров линейных регуляторов:

1. Для любого линейного регулятора используется двух-слойная нейронная сеть вида: входной слой – скрытый слой – выходной слой. В скрытом слое расположена сигмоидальная функция активации, а в выходном – линейная.

2. Количество входов НС определяется на основе анализа разностного уравнения настраиваемого линейного регулятора. НС не может располагать меньшим объемом информации, чем сам регулятор. В частности, для ПИД-регулятора определен набор из 5 нейронов. В выходном слое находится столько нейронов, сколько составляющих в настраиваемом регуляторе.

3. Для расчета количества нейронов скрытого слоя необходимо знать, сколько входов в НС и по скольким точкам ведется усреднение контролируемых параметров ОУ. Для расчета предлагается использовать формулу (11).

4. Для расчета дискретности вызова НС и времени задержки ее входных сигналов Δt необходимо разделить имеющееся время самого короткого в отслеживаемой картине заданий переходного процесса на количество нейронов в скрытом слое. Также в п. 5 получено аналитическое уравнение для расчета данного параметра, которое возможно применить при наличии оценок параметров модели ОУ, представленной в виде (1).

5. Весовые коэффициенты и смещения для скрытого слоя НС генерируются случайным образом в промежутке $[-1;1]$. Смещения выходного слоя приравниваются к нулю. Матрица весовых коэффициентов выходного слоя рассчитывается с помощью метода экстремального обучения, причем в качестве ожидаемых выходов НС выступают используемые до начала подстройки в линейном регуляторе параметры. Таким образом НС задается «начальное состояние» – в начальный момент она формирует на своих выходах такие значения параметров регулятора, которые использовались до ее подключения.

6. На вход НС поступают усредненные данные, полученные с ОУ. Перед непосредственной подачей на вход нейронов входного слоя они нормализуются в промежуток $[0;1]$.

7. В дальнейшем НС нейросетевого оптимизатора обучается в оперативном режиме по методу обратного распространения ошибки. Скорости обучения нейронов выходного слоя определяет база правил, исходя из ситуации на ОУ.

При этом использование НС для ПИД-регулятора для настройки П- или ПИ-регулятора возможно.

Далее приведены результаты применения нейросетевого оптимизатора на двух муфельных электронагревательных печах.

7. Исследование работы нейросетевого оптимизатора на реальном объекте управления

7.1. ОПИСАНИЕ КОНТУРОВ УПРАВЛЕНИЯ ПЕЧАМИ СНОЛ-1,6.2,5.1/11-И4 И СНОЛ 40/1200

В качестве ОУ для натуральных экспериментов были выбраны муфельные электронагревательные печи типа СНОЛ-1,6.2,5.1/11-И4 и СНОЛ 40/1200. Реализованная система управления такими печами, функциональная схема которой представлена на рис. 11, состояла из трех уровней.

На нижнем уровне находились термопара, измеряющая температуру рабочей зоны печи, и реле, посредством симисторов включающее нагревательный элемент.

Управляющий уровень (Simatic S7-300) осуществлял непосредственное управление муфельной электропечью. Контроллер получал данные от термопары через термопреобразователь, производил их обработку и отправлял на рабочую станцию, от которой он получал значение управляющего воздействия. В контроллере производилась широтно-импульсная модуляция управляющего воздействия и его выдача на исполнительный механизм включения нагревательного элемента.

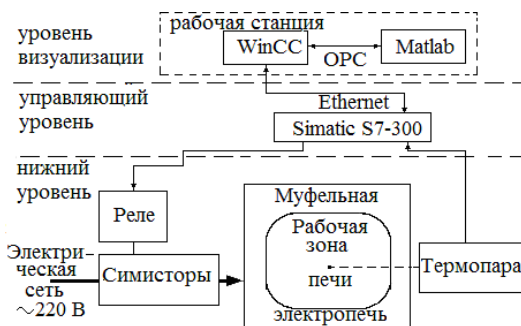


Рис. 11. Функциональная схема системы управления печью

На уровне визуализации находилась рабочая станция с системой визуализации WinCC и пакетом Matlab, в котором реализован нейросетевой оптимизатор параметров ПИ-регулятора. WinCC и Matlab взаимодействовали между собой через OPC-интерфейс посредством тегов.

Схема управления, реализованная в пакете Matlab, приведена на рис. 12. WinCC передавал данные о температуре (*temper*) и уставку (*zadanie*) по температуре в Matlab. Запущенная в Matlab HC формировала значение управляющего воздействия (*upravl*), которое через WinCC поступало в контроллер. *Group1*, *group2* – это OPC блоки соответственно для записи и чтения значений в тегах WinCC.

В Matlab реализован нейросетевой оптимизатор (см. блок *S-function neuc_PI*, рис. 12) параметров ПИ-регулятора (*PI*). Здесь U_i , U_p , U_{pi} – сигналы управления на выходе *i*-, *p*-каналов и всего ПИ-регулятора соответственно. Блок *satur* использован для ограничения сигнала управления с ПИ-регулятора до диапазона $[0; 100]$ единиц. U – управляющее воздействие, подаваемое на ОУ.

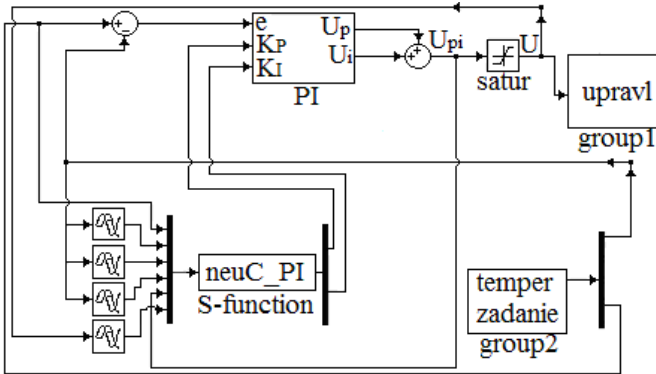


Рис. 12. Схема управления печью, реализованная в Matlab

Блок нейросетевого оптимизатора *S-function* имеет шесть входов. Первые пять из них являются входами НС (см. п. 4.5). Шестой вход необходим для корректной работы базы правил.

Для опытов использовался тот же график смены заданий, что и для модельных экспериментов: $590^{\circ}\text{C} \rightarrow 640^{\circ}\text{C} \rightarrow 505^{\circ}\text{C} \rightarrow 590^{\circ}\text{C}$ и т.д. Для смены уставки температура должна удерживаться в окрестности задания, равной 5% разницы между текущим заданием и предыдущим, в течение 30 минут.

Для пустых печей были подобраны коэффициенты ПИ-регулятора на переходном процессе $505^{\circ}\text{C} \rightarrow 590^{\circ}\text{C}$: 1) СНОЛ-1,6,2,5.1/11-И4: $K_p = 0,6$, $K_i = 6,976 \cdot 10^{-4}$, и 2) СНОЛ 40/1200: $K_p = 2,5$, $K_i = 1,67 \cdot 10^{-2}$. Для обеих печей в соответствии с подходом, предложенным в п. 4.6, было рассчитано значение Δt . Для СНОЛ -1,6,2,5.1/11-И4 $\Delta t = 20$ с, для СНОЛ 40/1200 $\Delta t = 10$ с.

7.2. РЕЗУЛЬТАТЫ НАТУРНЫХ ЭКСПЕРИМЕНТОВ

В рамках исследования для графика заданий $590^{\circ}\text{C} \rightarrow 640^{\circ}\text{C} \rightarrow 505^{\circ}\text{C} \rightarrow 590^{\circ}\text{C}$ для каждой из печей было проведено по два эксперимента: для системы с нейросетевым оптимизатором параметров ПИ-регулятора и для обычного ПИ-регулятора.

Эксперимент заключался в том, что исследовалась работа системы в следующих условиях: с начального момента и до момента, когда задание было изменено 12 раз, муфельная печь была пустой, далее в нее загружалась литая заготовка объемом примерно 30% от рабочего пространства печи, и задание изменялось еще 12 раз, а затем печь разгружалась и проводилось еще 12 переходных процессов. Каждый переходный процесс считался завершенным, если после входа в окрестность задания, равную 5% разницы между текущим заданием и предыдущим, проходило 30 минут. Затем происходила смена задания. Для этого был использован счетчик времени. Если температура в печи выходила за пределы окрестности, то счетчик времени обнулялся. Нейросетевой оптимизатор всегда начинал эксперимент с известными значениями коэффициентов ПИ-регулятора, указанными ранее как оптимальные для пустой печи. Обычный ПИ-регулятор всегда использовал только их.

Результаты экспериментов для печи СНОЛ –1,6.2,5.1/11-И4, полученные с использованием нейросетевого оптимизатора параметров ПИ-регулятора и обычного ПИ-регулятора, для графика смены заданий $590^{\circ}\text{C} \rightarrow 640^{\circ}\text{C} \rightarrow 505^{\circ}\text{C} \rightarrow 590^{\circ}\text{C}$ приведены на рис. 13 и рис. 14. Графики изменения параметров регулятора K_p и K_I также представлены на рис. 14.

До момента загрузки печи болванкой обе системы ведут себя практически одинаково. В период работы загруженной печи система с нейросетевым оптимизатором позволила добиться 4% перерегулирования для процессов нагрева и 2,2% – для охлаждений, в то время как для обычного ПИ-регулятора для процессов нагрева этот показатель составил 12%, для охлаждения – 4,5%. После разгрузки печи ПИ-регулятор с первого переходного процесса начинает работать как до загрузки печи. Системе с нейросетевым оптимизатором требуется некоторое время, чтобы вернуть параметры регулятора к исходным. Затем системы начинают работать практически идентично.

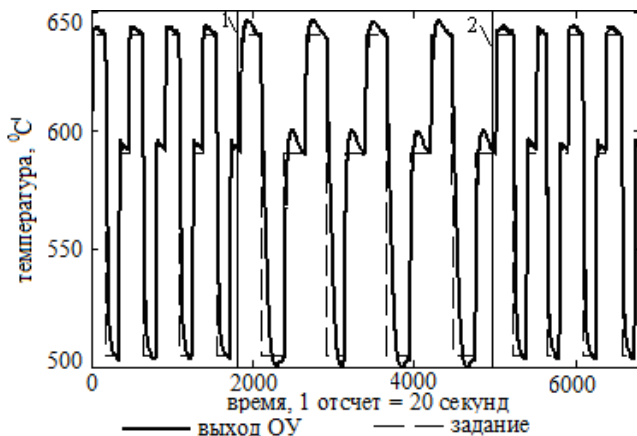


Рис. 13. Результаты для печи СНОЛ–1,6.2,5.1/11-И4 для графика изменения задания $590^{\circ}\text{C} \rightarrow 640^{\circ}\text{C} \rightarrow 505^{\circ}\text{C} \rightarrow 590^{\circ}\text{C}$ для обычного ПИ-регулятора. 1 – момент загрузки печи; 2 – момент разгрузки печи

Общее время опыта для системы с нейросетевым оптимизатором составило 29,08 часа при расходе электроэнергии 15,11 кВт·ч, а для обычного ПИ-регулятора – 37,7 часа при расходе электроэнергии 17,56 кВт·ч. Таким образом, экономия по времени на выполнение задания при использовании нейросетевого оптимизатора составила 23%, а электроэнергии – 14%.

Значение параметра K_I в определенные моменты (остывание печи) оказывалось равным нулю. Это связано со следующими причинами. Большую часть переходного процесса при остывании выход ПИ-регулятора является отрицательным. Допустимые пределы для сигнала управления – $[0; 100]$. Поэтому, в соответствии с п. 4.5, обучение сети не производится, а в соответствии с базой правил, K_I приравнивается к нулю.

Результаты экспериментов для печи СНОЛ 40/1200 приведены на рис. 15–16.

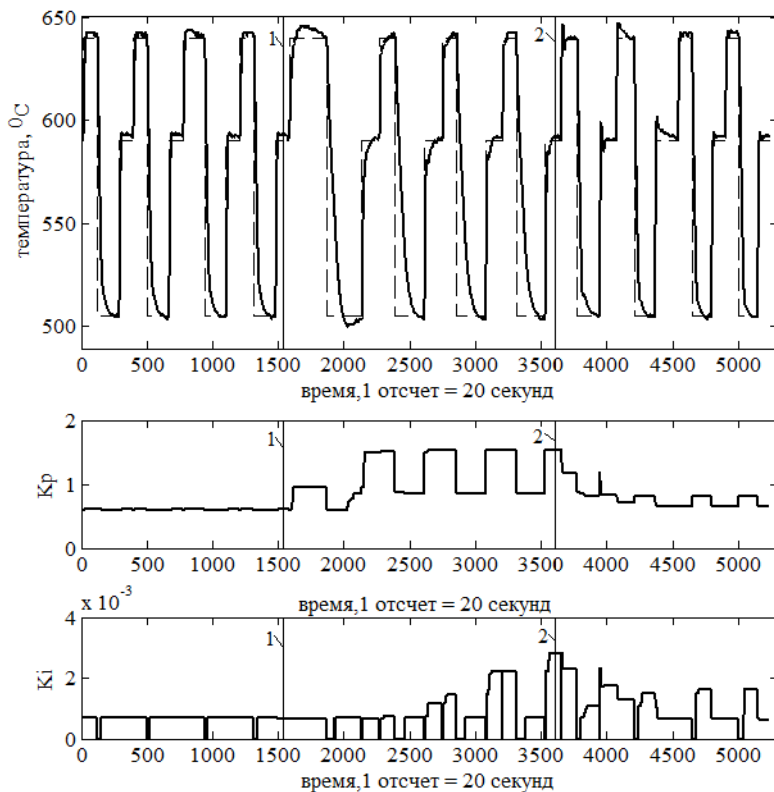


Рис. 14. Результаты для печи СНОЛ-1,6,2,5.1/11-И4 для графика изменения задания $590^{\circ}\text{C} \rightarrow 640^{\circ}\text{C} \rightarrow 505^{\circ}\text{C} \rightarrow 590^{\circ}\text{C}$ для нейросетевого оптимизатора. 1 – момент загрузки печи; 2 – момент разгрузки печи

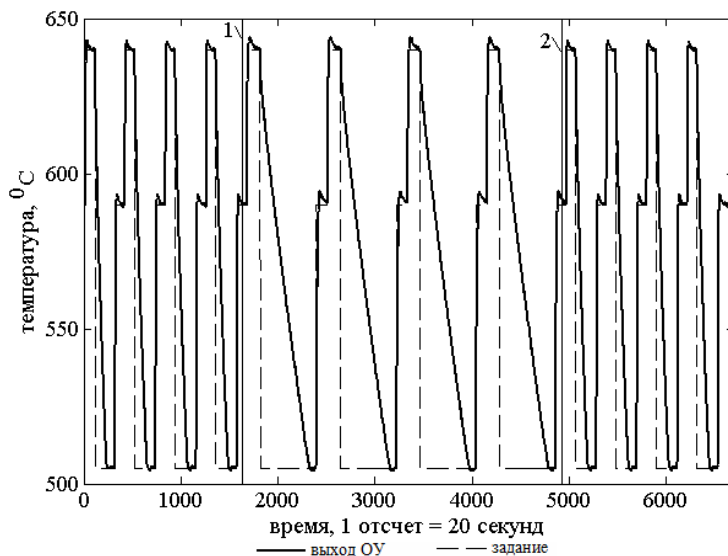


Рис. 15. Результаты системы управления с обычным ПИ-регулятором для печи СНОЛ 40/1200 для графика изменения задания $590^{\circ}\text{C} \rightarrow 640^{\circ}\text{C} \rightarrow 505^{\circ}\text{C} \rightarrow 590^{\circ}\text{C}$, 1 – момент загрузки печи; 2 – момент разгрузки печи.

В период работы загруженной печи система с нейросетевым оптимизатором позволила добиться 4% перерегулирования для процессов нагрева, в то время как для обычного ПИ-регулятора этот показатель составил 9%. Время опыта для системы с нейросетевым оптимизатором составило 35,42 часа при расходе электроэнергии 17,24 кВт·ч, а для обычного ПИ-регулятора – 36,87 часа при расходе 18,37 кВт·ч. Экономия по времени на выполнение графика задания при использовании нейросетевого оптимизатора составила 4%, а электроэнергии – 6%.

Подобные эксперименты были проведены для обеих печей на других графиках изменения задания:

1) $590^{\circ}\text{C} \rightarrow 620^{\circ}\text{C} \rightarrow 570^{\circ}\text{C} \rightarrow 590^{\circ}\text{C}$. В данном случае базовая температура была оставлена равной 590°C , но были уменьшены амплитуды ступенчатых изменений задания. Экономия по

времени относительно системы с обычным ПИ-регулятором составила: а) для печи СНОЛ-1,6.2,5.1/11-И4 – 22,8%, б) для печи СНОЛ 40/1200 – 4,6%. Экономия по расходу электроэнергии относительно системы с обычным ПИ-регулятором составила: а) для печи СНОЛ-1,6.2,5.1/11-И4 – 22,3%, б) для печи СНОЛ 40/1200 – 5,4%.

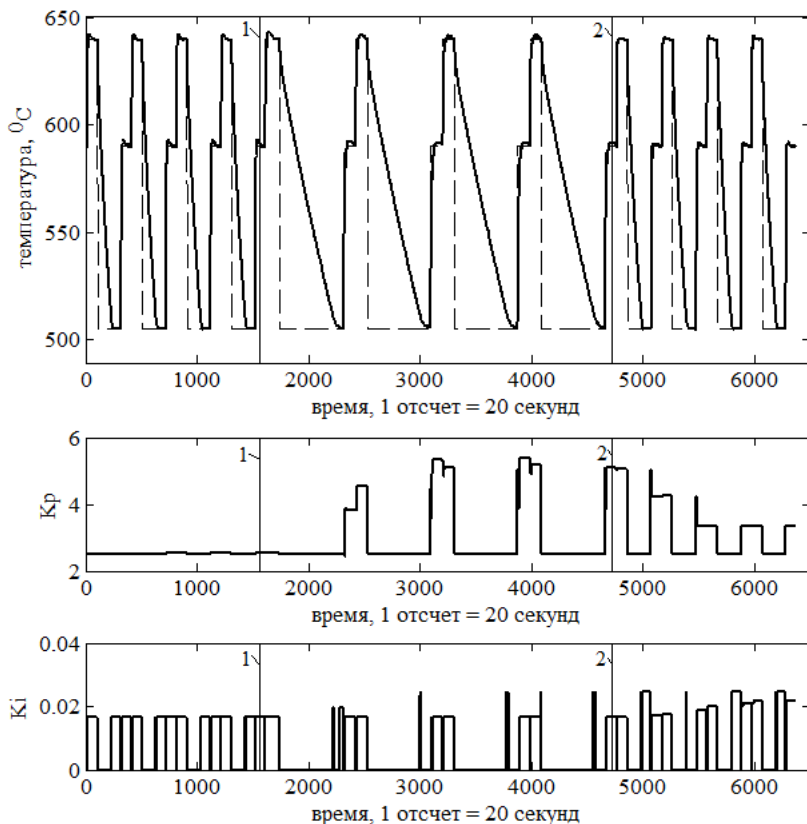


Рис. 16. Результаты применения нейросетевого оптимизатора для печи СНОЛ 40/1200 для графика изменения задания $590^{\circ}\text{C} \rightarrow 640^{\circ}\text{C} \rightarrow 505^{\circ}\text{C} \rightarrow 590^{\circ}\text{C}$, 1 – момент загрузки печи; 2 – момент разгрузки печи

2) $300^{\circ}\text{C} \rightarrow 330^{\circ}\text{C} \rightarrow 240^{\circ}\text{C} \rightarrow 300^{\circ}\text{C}$. В данном случае были изменены и базовая температура, и амплитуды ступенчатых изменений задания. Экономия по времени относительно системы с обычным ПИ-регулятором составила: а) для печи СНОЛ-1,6,2,5.1/11-И4 – 24%, б) для печи СНОЛ 40/1200 – 2,8%. Экономия по расходу электроэнергии относительно системы с обычным ПИ-регулятором составила: а) для печи СНОЛ-1,6,2,5.1/11-И4 – 21%, б) для печи СНОЛ 40/1200 – 5,2%.

3) $150^{\circ}\text{C} \rightarrow 190^{\circ}\text{C} \rightarrow 100^{\circ}\text{C} \rightarrow 150^{\circ}\text{C}$. В данном случае базовая температура вновь была уменьшена. Экономия по времени относительно системы с обычным ПИ-регулятором составила: а) для печи СНОЛ-1,6,2,5.1/11-И4 – 22%, б) для печи СНОЛ 40/1200 – 2,8%. Экономия по расходу электроэнергии относительно системы с обычным ПИ-регулятором составила: а) для печи СНОЛ-1,6,2,5.1/11-И4 – 31%, б) для печи СНОЛ 40/1200 – 3,2%.

4) $1000^{\circ}\text{C} \rightarrow 1050^{\circ}\text{C} \rightarrow 915^{\circ}\text{C} \rightarrow 1000^{\circ}\text{C}$. Подобный опыт проведен только для печи СНОЛ 40/1200. В данном случае температуры находятся около верхней допустимой границы для данной печи. Экономия по времени относительно системы с обычным ПИ-регулятором составила 4,7%, по расходу электроэнергии – 6,7%.

Полученные результаты подтверждают адекватность предложенного метода определения структуры и параметров НС нейросетевого оптимизатора.

Кроме того, необходимо отметить, что при подстройке параметров ПИ-регулятора при постоянном графике уставок и режиме работы печи параметры, K_P и K_I устанавливаются на некоторых постоянных значениях. Это позволяет сделать вывод о том, что оптимизатор может работать и в ином, относительно описанного, режиме. Если в момент его запуска параметры регулятора не позволяют получать требуемого качества переходных процессов, то он произведет их подстройку. После этого, при условии относительной стабильности параметров объекта управления, его возможно отключить. При следующей смене состояния объекта описанную процедуру возможно по-

вторить вновь. Если печь работает для строго определенного графика изменения уставок, объем и структура посадка материала в печь практически не изменяется, а список таких режимов ограничен, то со временем с помощью настройщика возможно подобрать параметры регулятора для всех таких режимов и перейти в режим табличной адаптации (gain scheduling [23]). Однако у непрерывной адаптации существует ряд преимуществ. Во-первых, она позволяет оперативно реагировать на дрейф характеристик объекта. Во-вторых, для реальных печей, например металлургического производства, в зависимости от марки металла меняется график уставок. И даже для одной марки объем посадка металла в печь может существенно отличаться. В-третьих, в случае, например, капитального ремонта печи для всех режимов работы объекта параметры регулятора придется уточнять.

8. Заключение

В статье предложен метод синтеза и инициализации нейронной сети для нейросетевого оптимизатора параметров линейных регуляторов. Для данного метода не требуется наличие модели ОУ, а достаточно лишь априорных сведений о нем: является ли ОУ симметричным или нет, какие значения параметров регулятора используются в настоящее время, какие значения критериев качества переходных процессов являются допустимыми, ограничения на значение управляющего воздействия. Метод может быть применен для линейного регулятора любой структуры, основанной на П-, И-, Д- составляющих. Изложены подходы к выбору количества слоев, нейронов в них, функций активации и времени задержки входных сигналов сети, являющихся выходными сигналами ОУ. Приведен разработанный общий алгоритм синтеза НС.

Предложенный метод синтеза и инициализации нейронной сети нейросетевого оптимизатора параметров линейных регуляторов проверен как на моделях объектов с различной динамикой, так и на реальных лабораторных печах.

Полученные результаты позволяют сделать вывод о том, что использование нейросетевого оптимизатора параметров линейных регуляторов с синтезированной нейронной сетью для учета нелинейных свойств ОУ позволило сократить перерегулирование и время, требуемое на выполнение графика заданий, а также существенно сократить расход электроэнергии. Таким образом, в условиях реального производства на данном агрегате может быть интенсифицировано производство продукции и сокращено удельное энергопотребление. Для всех опытов нейронная сеть имела одинаковую структуру. Так же как и база правил.

В дальнейшем предполагается провести модельные и натурные эксперименты по использованию нейросетевого оптимизатора для управления объектами с другими динамическими характеристиками, в частности, главным электроприводом двухвалковой прокатной клетки.

Работа выполнена при поддержке Российского фонда фундаментальных исследований (грант 15-07-0609215) и фонда Президента РФ (грант №14.У30.15.4865-МК)

Литература

1. АЛЕКСАНДРОВ А.Г. *Оптимальные и адаптивные системы*. – М.: Высшая школа, 1989. – 263 с.
2. АРНОЛЬД В.И. *О функциях трех переменных* // Доклады АН СССР. – 1957. – Т. 114, №4. – С. 679–681.
3. ВАСИЛЬЕВ В.И., ИДРИСОВ И.И. *Алгоритмы проектирования и анализа устойчивости интеллектуальной системы управления* // Вестник Уфимского государственного авиационного технического университета. – 2008. – Т. 11, №1. – С. 34–42.
4. ЕРЕМЕНКО Ю.И., ПОЛЕЩЕНКО Д.А., ГЛУЩЕНКО А.И. *О применении нейросетевого оптимизатора параметров ПИ-регулятора для управления нагревательными печами в различных режимах работы* // Управление большими системами. – 2015. – Вып. 56. – С. 143–175.

5. ЕРЕМЕНКО Ю.И., ПОЛЕЩЕНКО Д.А., ГЛУЩЕНКО А.И. *Об особенностях практической реализации схемы ПИД-нейрорегулятора с самонастройкой для управления печами нагрева* // Приборы и системы. Управление. Контроль. Диагностика. – 2012. – №1. – С. 25–30.
6. ЗМЕУ К.В., МАРКОВ Н.А., ШИПИТЬКО И.А. И ДР. *Безмодельное прогнозирующее инверсное нейроуправление с регенерируемым эталонным переходным процессом* // Информатика и системы управления. – 2009. – №3. – С. 109–117.
7. КОЛМОГОРОВ А.Н. *О представлении непрерывных функций нескольких переменных суперпозициями непрерывных функций меньшего числа переменных* // Доклады АН СССР. – 1956. – Т. 108, №2. – С. 179–182.
8. КРАСОВСКИЙ А.А., БУКОВ В.Н., ШЕНДРИК В.С. *Универсальные алгоритмы оптимального управления непрерывными процессами*. – М.: Наука, 1977. – 272 с.
9. КРУГЛОВ С.П. *Взаимосвязь двух подходов к аналитическому конструированию оптимальных регуляторов* // Автоматика и телемеханика. – 2003. – №4. – С. 56–69.
10. КУДИНОВ Ю.И., КЕЛИНА А.Ю. *Упрощенный метод определения параметров нечетких ПИД регуляторов* // Мехатроника, автоматизация, управление. – 2013. – №1. – С. 12–22.
11. ЛЕТОВ А.М. *Математическая теория процессов управления*. – М.: Наука, 1981. – 255 с.
12. ЛОВЧАКОВ В.И., СУХИНИН Б.В., СУРКОВ В.В. *Оптимальное управление электротехническими объектами*. – Тула: Издательство ТулГУ, 2004. – 149 с.
13. НАКОНЕЧНИЙ М.В., ГРНЯК Ю.Б., ІВАХІВ О.В. И ДР. *Дослідження стійкості системи з нейроконтролером* // Вісник НТУУ “КПІ”. Серія ПРИЛАДОБУДУВАННЯ. – 2014. – Вип. 48(2). – С. 139–148.
14. ОМОРОВ Т.Т., ДЖОЛДОШЕВ Б.О. *Краткий обзор методов анализа и синтеза нелинейных САУ* // Известия Кыргызского государственного технического университета им. И. Раззакова. – 2012. – №26. – С. 172–183.
15. РОТАЧ В.Я. КУЗИЩИН В.Ф. ПЕТРОВ С.В. *Настройка регуляторов по переходным характеристикам систем управления*

- без их аппроксимации аналитическими выражениями // Автоматизация в промышленности. – 2009. – № 11. – С. 9–12.
16. САЛИХОВ З.Г., АРУНЯНЦ Г.Г., РУТКОВСКИЙ А.Л. *Системы оптимального управления сложными технологическими объектами*. – М.: Теплоэнергетик, 2004. – 496 с.
 17. ФЕЛЬДБАУМ А.А. *Основы теории оптимальных автоматических систем*. – М.: Наука, 1971. – 744 с.
 18. ШУБЛАДЗЕ А.М., КУЗНЕЦОВ С.И. *Автоматически настраиваемые промышленные ПИ и ПИД регуляторы* // Автоматизация в промышленности. – 2007. – №2. – С. 15–17.
 19. ALEXANDROV A.G., PALENOV M.V. *Self-tuning PID-I controller* // Proc. 18th IFAC World Congress, Milano, Italy, 2011. – P. 3635–3640.
 20. ALLAOUA B., GASBAOUI B., MEBARKI B. *Setting up PID DC motor speed control alteration parameters using particle swarm optimization strategy* // Leonardo Electronic Journal of Practices and Technologies. – 2009. – Vol. 14. – P. 19–32.
 21. ANDERSON K.L., BLANKENSHIP G.I., LEBOW L.G. *A rule-based adaptive PID controller* // Proc. 27th IEEE Conf. Decision. Control, 1988. – P. 564–569.
 22. ARAI M. *Bounds on the number of hidden units in binary-valued three-layer neural networks* // Neural Networks. – 1993. – Vol. 6, No. 6. – P. 855–860.
 23. ASTROM K.J., HAGGLUND T. *Advanced PID Control*. – Research Triangle Park: ISA – The Instrumentation, Systems, and Automation Society, 2006. – 461 p.
 24. ASTROM K.J., HAGGLUND T., HANG C.C. et al. *Automatic tuning and adaptation for PID controllers. A survey* // IFAC J. Control Eng. Practice. – 1993. – Vol. 1, No. 4. – P. 699–714.
 25. BARTLETT E.B. *Dynamic node architecture learning: An information theoretic approach* // Neural Networks. – 1994. – No. 7. – P. 129–140.
 26. BAZYLEVYCH O., HIRNYAK Y., IVAKHIV O. *Neural controller for designing of nanopositioning systems* // IEEE International Workshop on Intelligent Data Acquisition and Advanced Compu-

- ting Systems: Technology and Applications. – Rende: IEEE, 2009. – P. 287–290.
27. BINDU R., NAMBOOTHIRIPAD M.K. *Tuning of PID controller for DC servo motor using genetic algorithm* // International Journal of Emerging Technology and Advanced Engineering. – 2012. – Vol. 2, No. 3. – P. 310–314.
 28. BIRBIR Y., NOGAY H.S., TOPUZ V. *Estimation of total harmonic distortion in short chorded induction motors using artificial neural network* // Proceedings of the 6th WSEAS Int. Conf. on Applications of Electrical Engineering. – Istanbul: Istanbul Technical University, 2007. – P. 206–210.
 29. CHEN J., HUANG T. *Applying neural networks to on-line updated PID controllers for nonlinear process control* // J. of Process Control. – 2004. – No. 14. – P. 211–230.
 30. CHEN J., YEA Y. *Neural network-based predictive control for multivariable processes* // Chemical Engineering Communications. – 2002. – Vol. 189. – No. 7. – P. 865–894.
 31. CHIEN K.L., HRONES I.A., RESWICK J.B. *On the automatic control of generalized passive systems* // Trans on ASME. – 1952. – Vol. 74. – P. 633–640.
 32. CYBENKO G. *Approximation by superposition of a sigmoidal function* // Math. Control Systems and Sigansls. – 1989. – No. 2. – P. 303–314.
 33. DEMIROREN A., YESIL E. *Automatic generation control with fuzzy logic controllers in the power system including SMES units* // International Journal of Electrical Power & Energy Systems. – 2004. – Vol. 26(4). – P. 291–305.
 34. ERENOGLU I., EKSIN I., YESIL E. AND ETC. *An intelligent hybrid fuzzy PID controller* // European Conference on Modelling and Simulation. – Bonn: European Council for Modelling and Simulation, 2006. – P. 62–67.
 35. FAHLMAN S.E., LEBIERE C. *The cascade-correlation learning architecture* // Advances in Neural Information Processing Systems. – 1990. – Vol. 2. – P. 524–532.
 36. FANG M., ZHUO Y., LEE Z. *The application of the self-tuning neural network PID controller on the ship roll reduction in random*

- waves // Ocean Engineering. – 2010. – No. 37. – P. 529–538.
37. FERREIRA P.M., FARIA E.A., RUANO A.E. *Neural network models in greenhouse air temperature prediction* // Neurocomputing. – 2002. – Vol. 43(1). – P. 51–75.
 38. FREAN M. *The upstart algorithm: A method for constructing and training feedforward neural networks* // Neural Computation. – 1990. – No. 2. – P. 198–209.
 39. FUNAHASHI K. *On the approximate realization of continuous mappings by neural networks* // Neural Networks. – 1989. – Vol. 2, No. 3. – P. 183–192.
 40. GOMM J.B., YU D.L., WILLIAMS D. *A new model structure selection method for non-linear systems in neural modeling* // UKACC International Conference on Control. – London: Institution of Electrical Engineers, 1996. – P. 752–757.
 41. GORBAN A.N., WUNSCH D. *The general approximation theorem* // Proceedings of the IJCNN. – Anchorage: IEEE, 1998. – P. 1271–1274.
 42. GUZELKAYA M., EKSIN I., YESIL E. *Self-tuning of PID-type fuzzy logic controller coefficients via relative rate observer* // Engineering Applications of Artificial Intelligence. – 2003. – Vol. 16(3). – P. 227–236.
 43. HAGGLUND T., TENGVALL A. *An automatic tuning procedure for unsymmetrical processes* // Proc. 3rd European control conference, 1995. – P. 2450–2455.
 44. HECHT-NIELSEN R. *Kolmogorov's Mapping Neural Network Existence Theorem* // Proc. Int. Conf. Neural Networks. – New York: IEEE Press, 1987. – Vol. III. – P. 11–14.
 45. HORNICK K., STINCHCOMBE M., WHITE H. *Multilayer feedforward networks are universal approximators* // Neural Networks. – 1989. – Vol. 2, No. 5. – P. 359–366.
 46. HUANG G.-B. *Learning capability of neural networks* // Ph.D. Thesis. – Singapore: Nanyang Technological University, 1998. – 137 p.
 47. HUANG G.-B., BABRI H.A. *Upper bounds on the number of hidden neurons in feedforward networks with arbitrary bounded*

- nonlinear activation functions* // IEEE Trans. Neural Networks. – 1998. – Vol. 9(1). – P. 224–229.
48. HUANG G.-B., WANG D.H., LAN Y. *Extreme learning machines: a survey* // International Journal of Machine Learning and Cybernetics. – 2011. – Vol. 2, No. 2. – P. 107–122.
49. HUANG G.-B., ZHU Q.Y., SIEW C.K. *Extreme learning machine: theory and applications* // Neurocomputing. – 2006. – Vol. 70, No. 1. – P. 489–501.
50. KALMAN R.E. *Theory of Optimal Control* // Bullet Soc. Mat. Mech. – 1960. – Vol. 5, No. 1. – P. 106–115.
51. KIM S.-S. *Time-delay recurrent neural network for temporal correlations and prediction* // Neurocomputing. – 1998. – Vol. 20. – P. 253–263.
52. KOLMOGOROV A.N. *On the representation of continuous function of many variables by superpositions of continuous functions of one variable and addition* // Doklady Akademii Nauk USSR. – 1957. – Vol. 114(5). – P. 953–956.
53. KUHN U. *A practice oriented setting rule for pid-controllers: The t-sum-rule* // Automatisierungstechnische Praxis. – 1995. – Vol. 37, No. 5. – P. 10–16.
54. KŮRKOVÁ V. *Kolmogorov's theorem is relevant* // Neural Computation. – 1991. – Vol. 3. – No. 4. – P. 617–622.
55. LEVA A., COX C., RUANO A.E. *Hands-on PID autotuning: a guide to better utilisation* // IFAC Professional Brief. – Laxenburg: IFAC, 2002. – 84 p.
56. LI Y., ANG K., CHONG C. *Patents, software, and hardware for PID control – an overview and analysis of the current art* // IEEE Control Systems Magazine. – 2006. – No. 26(1). – P. 42–54.
57. LIU Y., STARZYK J.A., ZHU Z. *Optimizing number of hidden neurons in neural networks* // Artificial Intelligence and Applications. – 2007. – No. 2. – P. 138–143.
58. OMATU S., KHALID M., YUSOF R. *Neuro-Control and its Applications*. – London: Springer, 1995. – 255 p.
59. PFEIFFER B.-M. *Towards «plug and control»: self-tuning temperature controller for PLC* // Int. J. of Adaptive Control and Signal Processing. – 2000. – No. 14. – P. 519–532.

60. REFENES A.N., CHAN E.B. *Sound recognition and optimal neural network design* // Microprocessing and Microprogramming. – 1992. – Vol. 35, No. 1. – P. 783–789.
61. REYES J., ASTORGA C., ADAM M. AND ETC. *Bounded neuro-control position regulation for a geared DC motor* // Engineering Applications of Artificial Intelligence. – 2010. – No. 23. – P. 1398–1407.
62. RUANO A.E., CRISPIM E.M., CONCEICAO E.Z.E. AND ETC. *Prediction of building's temperature using neural networks models* // Energy and Buildings. – 2006. – Vol. 38(6). – P. 682–694.
63. SCHAEDEL H.M. *A new method of direct PID controller design based on the principle of cascaded damping ratios* // European Control Conference (ECC). – IEEE, 1997. – P. 1265–1271.
64. SHEELA K.G., DEEPA S.N. *Review on methods to fix number of hidden neurons in neural networks* // Mathematical Problems in Engineering. – 2013. – Vol. 2013. – P. 1–11.
65. STONE M.N. *The generalized Weierstrass approximation theorem* // Math. Mag. – 1948. – Vol. 21. – P. 167–183, 237–254.
66. SWINGLER K. *Applying neural networks: a practical guide*. – Waltham: Morgan Kaufmann, 1996. – 317 p.
67. TAMURA S., TATEISHI M. *Capabilities of a four-layered feed-forward neural network: four layers versus three* // IEEE Trans. Neural Networks. – 1997. – Vol. 8(2). – P. 251–255.
68. TAN S.-H., HANG C.-C., CHAI J.-S. *Gain scheduling: from conventional to neuro-fuzzy* // Automatica. – 1997. – Vol. 33, No. 3. – P. 411–419.
69. TRENN S. *Multilayer perceptrons: approximation order and necessary number of hidden units* // IEEE Transactions on Neural Networks. – 2008. – Vol. 19, No. 5. – P. 836–844.
70. UNAL M., AK A., TOPUZ V. AND ETC. *Optimization of PID Controllers Using Ant Colony and Genetic Algorithms*. – London: Springer, 2013. – 85 p.
71. UNAL M., ERDAL H., TOPUZ V. *Trajectory tracking performance comparison between genetic algorithm and ant colony optimization for PID controller tuning on pressure process* // Computer

- Applications in Engineering Education. – 2012. – Vol. 20(3). – P. 518–528.
72. UPADHYAYA B.R., ERYUREK E. *Application of neural networks for sensor validation and plant monitoring* // Neural Technology. – 1992. – Vol. 97. – P. 170–176.
 73. WEIGEND A.S., HUBERMAN B.A., RUMMELHART D.E. *Predicting sunspots and exchange rates with connectionist networks* // Nonlinear Modeling and Forecasting, SFI Studies in the Sciences of Complexity. – New York: Addison-Wesley, 1992. – Vol. 12. – P. 395–432.
 74. WEINZIERL K. *Method for generating control parameters from a response signal of a controlled system and system for adaptive setting of a PID controller* // Patent US 6353766. – 2002.
 75. XIA C. et al. *Speed control of brushless DC motor using genetic algorithm based fuzzy controller* // Proc. 2004 International Conference on Intelligent Mechatronics and Automation, Chengdu, China, 3rd edn. A Treatise on Electricity and Magnetism. – Chengdu: IEEE, 2004. – Vol. 2. – P. 68–73.
 76. YESIL E., GUZELKAYA M., EKSIN I. *Self tuning fuzzy PID type load and frequency controller* // Energy Conversion and Management. – 2004. – Vol. 45(3). – P. 377–390.
 77. YU D.L., GOMM J.B., WILLIAMS D. *Neural model input selection for a MIMO chemical process* // Engineering Applications of Artificial Intelligence. – 2000. – Vol. 13. – P. 15–23.
 78. YUAN H.C., XIONG F.L., HUAI X.Y. *A method for estimating the number of hidden neurons in feed-forward neural networks based on information entropy* // Computers and Electronics in Agriculture. – 2003. – Vol. 40, No. 1. – P. 57–64.
 79. ZHAO Z.Y., TOMIZUKA M., ISAKA S. *Fuzzy gain scheduling of PID controllers* // IEEE Transactions on systems, man, and cybernetics. – 1993. – Vol. 23, No. 5. – P. 1392–1398.
 80. ZIEGLER J., NICHOLS N. *Optimum settings for automatic controllers* // Trans. ASME. – 1942. – No. 65. – P. 759–768.

NEURAL NETWORK STRUCTURE SELECTION METHOD TO SOLVE LINEAR CONTROLLERS PARAMETERS ADJUSTMENT PROBLEM

Yuri Eremenko, Sary Oskol technological institute n.a. A.A. Ygarov (branch) NUST "MISIS", Sary Oskol, Doctor of Science, professor (erem49@mail.ru).

Anton Glushchenko, Sary Oskol technological institute n.a. A.A. Ygarov (branch) NUST "MISIS", Sary Oskol, Cand.Sc (Sary Oskol, Makarenko microdistrict, 42, (4725)45-12-17, strondudd@mail.ru).

Abstract: The problem of a neural network structure selection is considered. It is used as a part of a neural tuner to adjust P-, PI- or PID-controllers parameters online to control nonlinear plants. A method to find a number of network layers, neurons in each of them, choose activation functions and calculate delay time for network delayed inputs is proposed. Such method does not need a plant model. An algorithm to synthesize and initialize the neural network for the neural tuner with the help of a-priori known data about the plant is developed. Having made the experiments with plant models and two electroheating furnaces, we conclude that neural tuner helps to achieve both time and energy consumption decrease to complete setpoint schedule in comparison with conventional linear controller. This fact shows that proposed method is valid.

Keywords: neural network, adaptive control, PID-controller, neural tuner, neural network structure selection, input signals delay time.

*Статья представлена к публикации
членом редакционной коллегии П.С. Щербаковым.*

*Поступила в редакцию 20.01.2016.
Опубликована 31.07.2016.*