

С. В. Феофилов, д-р техн. наук, проф., svfeofilov@mail.ru,

А. В. Козырь, канд. техн. наук, доц., kozyr_a_v@mail.ru, Д. Л. Хапкин, мл. науч. сотр., dima-hapkin@ya.ru,
Тулский государственный университет

Структурно-параметрический синтез нейросетевых регуляторов для объектов управления с ограничителями*

Излагается методика синтеза цифровых систем управления для нелинейных объектов с ограничителями в условиях неполной информации. Рассматриваются замкнутые следящие системы с отрицательной обратной связью. Для построения регулятора, который включается последовательно с объектом управления, предлагается использовать искусственные нейронные сети.

Такой подход эффективен, когда известные классические методы не позволяют напрямую синтезировать управление. Это происходит, например, в случае, если математическая модель является существенно нелинейной и не полностью определена. Разработанные методы позволяют расширить класс технических систем, для которых возможен прямой (без использования различного рода упрощений) синтез близких к оптимальным законов управления. Кроме того, нейросетевые регуляторы обладают свойствами робастности, адаптивности, являются исходно цифровыми, т. е. имеют те качества, которые очень востребованы на практике. В статье основное внимание уделяется таким проблемам, как выбор структуры нейросети для нейроимитатора и нейрорегулятора, построение обучающей выборки, обеспечение сходимости процесса корректировки весов. Для обучения нейросетей в качестве базового используется метод обратного распространения ошибки.

Следует отметить, что сегодня нейросетевые технологии достаточно широко распространены в различных сферах деятельности. Впечатляют успехи, продемонстрированные в области обработки звука, изображения, автоматического перевода, в системах навигации, при обработке больших данных. Однако их применение в системах автоматического управления не столь широко. Авторы статьи считают, что потенциал искусственных нейронных сетей может быть использован в данном направлении. При этом следует понимать, что применение нейросетей эффективно лишь при определенных условиях и свойствах объекта управления.

Ключевые слова: нейросетевой регулятор, следящая система, обучающая выборка, нейроимитатор

Введение

В настоящее время при проектировании современных систем автоматического управления (САУ) для различных технических устройств имеется несколько основных тенденций. Во-первых, постоянно растет сложность объектов управления (ОУ) и их математических моделей. Это происходит благодаря усложнению самих технических объектов, связанному с использованием новых материалов, технологий и т. д., а также рассмотрению новых ранее недоступных режимов работы (например, гиперзвуковых скоростей в ракетостроении). Во-вторых, при практическом проектировании перспективных изделий нередко отсутствует полная, точная и верифицированная математическая модель объекта управления. В некоторых слу-

чаях получение такой модели вообще невозможно. В-третьих, подавляющее большинство САУ реализуется на цифровых вычислителях. Это вполне логично и является следствием их многочисленных преимуществ. Однако появляется дискретность сигналов по времени и уровню, что приводит к известным теоретическим и практическим проблемам (появляются дополнительные нелинейности и задержки, возможны автоколебания и т. д.). Далее, неточность модели, технологический разброс параметров, а также возможность их изменения в процессе работы приводит к нестационарности объекта. Это значит, что система управления должна обладать свойствами робастности и адаптивности. При этом, как правило, требуется получение законов управления, близких к оптимальным по заданному критерию.

Конечно, существуют известные классические методы синтеза, но их возможности существенно ограничены, особенно для сложных

*Исследование выполнено за счет гранта Российского научного фонда № 23-29-00609, <https://rscf.ru/project/23-29-00609/>.

нелинейных объектов управления. Например, при описанных выше условиях пришлось бы одновременно применять методы теории нелинейных цифровых систем, адаптивных и робастных систем, методы синтеза нестационарных и при этом оптимальных систем управления. Очевидно, что построить по этому принципу работоспособную методику синтеза не представляется возможным. Поэтому актуальным как с теоретической, так и с практической точки зрения является поиск подходов, алгоритмов и методик, способных в некоторых случаях справиться с обозначенной проблемой. Авторы считают, что при этом могут быть использованы методы искусственного интеллекта, в частности, нейросетевой подход, который позволяет использовать вычислительные возможности современных компьютеров для реализации перспективных численных алгоритмов.

Вообще, задача синтеза и оптимизации регуляторов для нелинейных систем управления является предельно общей и составляет основное содержание современной теории управления. Поэтому следует определиться с классом объектов управления и типом регулятора. Достаточно масштабной является задача построения цифровых следящих систем управления, основанных на искусственных нейронных сетях (ИНС), для нелинейных объектов управления с ограничителями при недостатке или неточности информации. Объекты указанного класса очень часто встречаются в технических системах. Они присутствуют, например, в электрических, гидравлических и газовых приводах. Если рассматривать объект управления как совокупность типовых структур, то исходя из математического описания звеньев, содержащих ограничители, можно выделить два типа последних: ограничители в форме механических упоров и в форме насыщения. Звено, содержащее ограничители первого типа, задается дифференциальным уравнением второго порядка с разрывной правой частью, причем разрывной (из-за удара об упор) является и фазовая траектория звена. Звено с ограничителями второго типа описывается дифференциальным уравнением первого порядка с разрывной правой частью, а его траектория является непрерывной функцией.

Выбор нейросетевого регулятора обусловлен тем, что при синтезе реальных технических систем математическая модель является неточной, присутствует значительный технологический разброс параметров, невозможно измерять

и использовать для управления весь вектор состояния. Все это приводит к ухудшению качества законов управления, а значит и конкретных характеристик готового изделия. ИНС выгодно отличается тем, что может быть обучена с использованием только входного и выходного сигналов как на математической модели, так и на реальном объекте. При этом ИНС обладает свойствами адаптивности и робастности.

Анализ применения ИНС в системах автоматического управления

Традиционно ИНС применяются в таких областях, как компьютерное зрение, машинное обучение, виртуальная реальность. Впечатляют достижения при использовании ИНС для анализа и синтеза фото- и видеоизображений, звука, в системах автоматического перевода, банковской сфере и т. д. В последнее время нейронные сети широко применяются для решения обширного круга задач робототехники и мехатроники. Применение глубокого обучения в робототехнике приводит к очень специфическим проблемам и исследовательским вопросам, которые не типичны и не рассматриваются в указанных выше приложениях. Основные современные исследования в области применения нейронных сетей и глубокого обучения [1] к задачам управления сложными динамическими системами можно подразделить на следующие направления.

1. Синтез робастных нейросетевых регуляторов. Важным направлением в современной теории управления является синтез робастного (нечувствительного к параметрическим отклонениям объекта) управления. Одним из наиболее мощных подходов к управлению не полностью определенными системами с ограничениями является методика, использующая понятие инвариантного множества управления, для которого существуют доказуемые гарантии существования допустимого управления с обратной связью, способная обеспечить нахождение любой замкнутой траектории системы в заданном множестве допустимых состояний [2–5]. Традиционно это достигается путем связывания управляющей функции Ляпунова с планом инвариантного множества. Новый подход к решению таких задач состоит в использовании нейрорегуляторов. Следует отметить работу [6], в которой рассматривается методика синтеза на основе аппроксимации традиционных стаби-

лизирующих регуляторов для линейных систем с политопической неопределенностью, в том числе регуляторов с переменной структурой. Архитектура нейронной сети строится на основе функции активации типа ReLU.

2. Идентификация неопределенностей в процессе функционирования. Распространенным допущением в глубоком обучении нейросетевого регулятора является то, что обученные модели реализуется в условиях закрытого множества [3], т. е. классы сигналов, встречающиеся во время эксплуатации, известны и являются точно такими же, как и во время обучения. Однако объекту управления часто приходится работать в постоянно меняющихся, неконтролируемых условиях реального мира, и они неизбежно сталкиваются с экземплярами классов, сценариев или условий окружающей среды, которые не были охвачены обучающими данными. Исследованию проблем идентификации неопределенностей посвящены работы [7–9].

3. Выбор архитектуры нейронной сети для нейрорегулятора. Основными архитектурами ИНС для применения в нейрорегуляторах считаются многослойные, рекуррентные и радикально-базисные нейронные сети. К настоящему времени разработаны типовые схемы САУ на основе нейрорегуляторов [9]. Эти схемы используют в себе только архитектуру многослойного перцептрона с временными задержками на входе [10]. Каждая из типовых схем имеет свои недостатки и не является универсальной. Разработчику нейросетевого регулятора придется провести исследование для требуемого объекта управления для выбора схемы включения регулятора в систему управления и выбора структуры используемой нейронной сети.

В результате революции глубокого обучения появились новые функции активации, изменены и доработаны методы обучения нейронных сетей, появились новые архитектуры нейронных сетей. Также активно ведутся исследования в области обучения с подкреплением с использованием глубоких ИНС [9, 11]. При таком методе ИНС учится взаимодействовать с окружающей средой (виртуальной или реальной), получая от нее обратный отклик о своих действиях (имеет обратную связь), для достижения поставленной цели. Данные исследования ведутся в основном для обучения поведения мультиагентных систем.

4. Задачи оптимизации. Существует также большой объем работ по использованию нейрон-

ных сетей для непосредственного решения задач нелинейного программирования [12]. Основная идея этих методов основана на использовании ИНС, моделирующей обратную динамику объекта, что позволяет свести задачу нелинейного программирования с ограничениями к задаче без ограничений и дает возможность существенно ускорить процедуру оптимизации. В работе [13] предлагается метод последовательного квадратичного программирования, использующий модель нейронной сети Холфилда. Сеть рассматривается как динамическая система, сходящаяся при времени, стремящемся к бесконечности, к оптимальному решению. Известны работы, например [14], в которых нейронная сеть используется для аппроксимации оптимального регулятора (нейронная динамическая оптимизация). В автономном режиме происходит настройка нейронной сети в соответствии с каким-либо показателем качества. Далее нейронную сеть необходимо переобучать в зависимости от параметрических изменений объекта управления. Обученная нейронная сеть затем используется в реальном масштабе времени. Также известны работы, в которых нейросети аппроксимируют таблично заданные нелинейные функции с целью получить дифференцируемую функцию для процедуры оптимизации. В работах показано, что нейронная сеть позволяет увеличить гладкость полученных решений по сравнению с использованием сплайнов.

Недавние исследования показывают, что глубокое обучение может быть эффективным инструментом для разработки оптимальных систем управления для многомерных нелинейных динамических систем. Но поведение этих нейросетевых контроллеров до сих пор не совсем изучено. Разработка оптимальных контроллеров с обратной связью для многомерных нелинейных систем остается сложной и нерешенной проблемой. Даже если предположить, что динамика системы полностью известна, то для проектирования таких регуляторов необходимо решить задачу Гамильтона—Якоби—Беллмана (PDE), размерность которой такая же, как у пространства состояний. Это приводит к известной проблеме "проклятия размерности", исключающей традиционные подходы, основанные на дискретизации. Некоторые авторы [15–17] используют интерполяцию с разреженной сеткой для аппроксимации решения уравнения Гамильтона—Якоби—Беллмана, называемого функцией ценности, и его градиента, который использу-

ется для вычисления оптимального управления с обратной связью. Это направление получило дальнейшее развитие с использованием нелинейной регрессии [11, 18] и разреженных полиномов, что значительно увеличило максимально возможную размерность задачи. В качестве альтернативы можно напрямую аппроксимировать градиент значений [9, 17].

5. Устойчивость нейросетевого управления.

Несмотря на многочисленные примеры успешного применения ИНС в задачах управления динамическими системами, нейросетевые системы управления все еще не имеют теоретических обоснований (гарантий) их устойчивости, что может препятствовать их применению во многих критически важных областях, связанных с безопасностью. Основное объяснение этому кроется в их репутации "черных ящиков", поведение которых невозможно предсказать. Существующие подходы к оценке обученных моделей в основном основаны на тестировании с использованием наборов данных. Однако тестирование показывает наличие, а не отсутствие ошибок. Если модели глубокого обучения будут использоваться, например, в таких приложениях, как управление автономными летательными аппаратами, необходимо иметь возможность проверять критически важное для безопасности поведение замкнутой системы.

Важнейшим критерием, который в настоящее время не может быть проверен для нейросетевых регуляторов, является оценка устойчивости замкнутой системы. В большинстве современных работ делаются попытки использования прямого метода Ляпунова. Существуют различные подходы к численному построению функций Ляпунова для нелинейных систем с нейросетевым управлением. Многие из них основаны на полиномиальной аппроксимации динамики и поиске суммы квадратов полиномов в качестве функций Ляпунова с помощью полуопределенного программирования (SDP) [19]. Такой подход исследовался в большом числе теоретических работ, однако на практике полиномиальные приближения накладывают значительные ограничения на системы и структуру функций Ляпунова. Более того, хорошо известны проблемы численной чувствительности в SDP [20], что делает поиск решений, полностью удовлетворяющих условиям Ляпунова, очень трудным.

Таким образом, исследования в области нейросетей ведутся очень активно, получены

многочисленные теоретические и практические результаты. Благодаря этому нейросети широко используются во многих областях. При этом их применение в качестве регуляторов в системах управления ограничено. Авторы считают, что следует восполнить этот пробел и использовать потенциал нейросетей для управления техническими объектами в условиях неполной информации.

Структурный синтез замкнутой нейросетевой системы управления

Далее будем рассматривать ИНС прямого распространения, для обучения которых применим метод обратного пропуска ошибки через прямой нейроимитатор [9, 21]. В этом методе используются две нейронные сети, одна из которых выполняет функцию регулятора, а вторая — модели объекта управления, которая называется прямым нейроэмулятором. Схемы обучения и управления представлены на рис. 1 (см. вторую сторону обложки).

Таким образом, первой задачей является выбор структуры и обучение нейроимитатора, который должен эмулировать поведение объекта управления. В общем виде структура многослойной ИНС показана на рис. 2.

Обучение ИНС — это процесс корректировки весовых коэффициентов на каждом входе нейрона w_j в целях обеспечения заданных критериев на выходе.

Сегодня подавляющее большинство ИНС реализуются на цифровых устройствах, и по этой причине будем рассматривать их в дискретном виде. Для линейного объекта из разностного уравнения можно получить зависимость для прямой динамики:

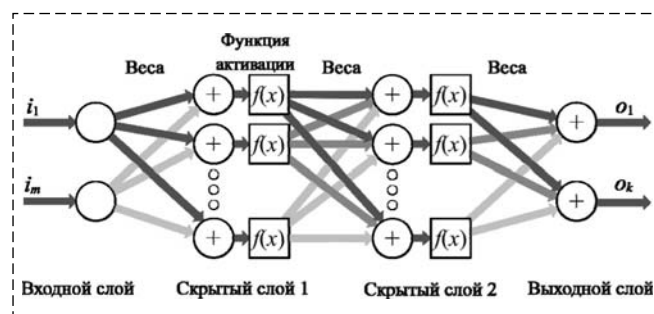


Рис. 2. Структура многослойной нейронной сети прямого распространения с двумя скрытыми слоями

Fig. 2. Structure of a multilayer forward propagation neural network with two hidden layers

$$y(i) = b_0 u(i) + b_1 u(i-1) + \dots + b_M u(i-M) - a_1 y(i-1) - \dots - a_K y(i-K) \quad (1)$$

где a_k и b_m — коэффициенты числителя и знаменателя Z -передаточной функции соответственно, $k = 0, 1, \dots, K$, $m = 0, 1, \dots, M$, $a_0 = 1$; u — сигнал на входе объекта управления; y — выход объекта управления (один из элементов вектора состояния x); i — текущий такт квантования.

Для регулятора, основанного на инверсной динамике, из уравнения (1) выражается $u(i-1)$, выполняется замена i на $(i+1)$ и $y(i+1)$ на $r(i)$:

$$u(i) = [r(i) + a_1 y(i) + \dots + a_K y(i-K+1) - b_2 u(i-1) - \dots - b_M u(i-M+1)]/b_1, \quad (2)$$

где r — полезный входной сигнал.

Таким образом, для линейных систем структуру нейроимитатора и нейрорегулятора можно выбирать, основываясь на выражениях (1) и (2) соответственно, т. е. это один нейрон без функции активации с весами, соответствующими коэффициентам разностного уравнения.

Далее рассматриваются ОУ с такими ограничителями, как ограничители типа насыщение (рис. 3) и типа жесткий механический упор (рис. 4).

Звено первого типа описывается следующим образом:

$$\dot{x} = \begin{cases} \frac{ku - x}{T}, & \text{если } |x| < D \text{ или } |x| = D \text{ и } \text{sign}(x) \leq 0; \\ 0, & \text{если } |x| = D \text{ и } \frac{ku - x}{T} > 0. \end{cases} \quad (3)$$

Звено с ограничителем типа жесткий механический упор имеет следующее описание:

$$\begin{cases} \dot{x}_1 = x_2; \\ \dot{x}_2 = \begin{cases} \frac{ku - x_1}{T^2} - \frac{2\xi x_2}{T}, & \text{если } |x_1| < D \\ \text{или } |x_1| = D \text{ и } \text{sign}(x_1) \leq 0; \\ 0, & \text{если } |x_1| = D \\ \text{и } (ku - x_1)\text{sign}(x_1) > 0. \end{cases} \end{cases} \quad (4)$$

При этом предполагается, что удар об упор является абсолютно неупругим, и сход с ограничителя является непрерывным:

$$\begin{cases} x_1(t^* + 0) = x_1(t^* - 0), \\ x_2(t^* + 0) = 0. \end{cases} \quad (5)$$

Для имитации таких объектов и управления ими предлагается использовать нейрон-

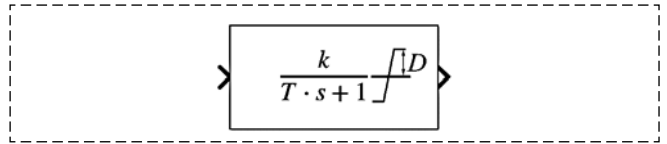


Рис. 3. Звено с ограничителем типа насыщение
Fig. 3. Saturation-type limiter link

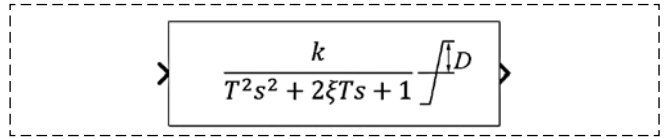


Рис. 4. Звено с ограничителем типа жесткий механический упор
Fig. 4. Link with limiter type rigid mechanical stop

ные сети с функцией активации ReLU [22], которая описывается следующим уравнением:

$$f(x) = \begin{cases} x, & x \geq 0; \\ 0, & x < 0. \end{cases} \quad (6)$$

На основе уравнений (3)—(6) авторами разработаны нейросетевые структуры, повторяющие дискретную динамику объектов с ограничителями. Например, для звена с ограничителем типа жесткий механический упор такая структура показана на рис. 5 (см. вторую сторону обложки).

На рис. 5 a_1 , a_2 , b_1 , b_2 — значения коэффициентов разностного уравнения (1), u_b и l_b — значение верхнего и нижнего ограничения, соответственно.

Обобщение приводит к следующей зависимости для минимального размера скрытого слоя нейроимитатора:

$$n_{\text{hidden}} = 4n_{\text{sat}} + 2n_{\text{lin}}, \quad (7)$$

где n — порядок системы, $n = n_{\text{sat}} + n_{\text{lin}}$, n_{sat} — число ограниченных фазовых переменных вектора состояния, n_{lin} — число неограниченных фазовых переменных вектора состояния.

В зависимости от возможности измерения фазовых переменных структуру нейронной сети можно формировать либо на основе обратных связей, составляющих полный вектор состояния объекта управления, либо на основе задержек, введенных в обратную связь по регулируемой величине. В первом случае минимальная структура сети имеет вид:

- 1-й слой (входной) — размерностью $n + 1$ (вектор состояния объекта и сигнал управления);
- 2-й слой (скрытый) — размерностью $4n_{\text{sat}} + 2n_{\text{lin}}$;
- 3-й слой (выходной) — размерностью n (вектор состояния на новом шаге).

При использовании задержек выходного сигнала:

- 1-й слой (входной) — размерностью $2n$ (состоит из сигнала управления объектом и $(n - 1)$ его задержанных значений, сигнала обратной связи и $(n - 1)$ ее задержанных значений);
- 2-й слой (скрытый) — размерностью $4n_{\text{sat}} + 2n_{\text{lin}}$;
- 3-й слой (выходной) — скалярный выход системы.

На рис. 6, 7 показаны результаты сравнения динамики звена с ограничителем типа жесткий механический упор с нейроимитатором при работе в линейной зоне и с достижением ограничителей соответственно.

На рис. 6 и 7 штриховой линией обозначен выход нейроимитатора, сплошной серой линией показан выход объекта управления. Из рис. 6, 7 видно, что имитатор с предложенной структурой отлично повторяет динамику объекта.

Универсальное описание структуры нейрорегулятора для произвольного объекта найти невозможно. В общем случае для ИНС прямого распространения требуется $2n$ входов, где n — порядок ОУ. Кроме того, предлагаются следующие соображения: порядок объекта определяет минимальный размер входа и число задержек для обратных связей, а для скрытых слоев следует

пользоваться правилом поэтапного увеличения нейронов и числа слоев до достижения результата, соответствующего заданным требованиям.

Формирование обучающей выборки для нейрорегуляторов в замкнутых системах с ограничителями

Для обучения нейрорегуляторов будет использован метод обратного распространения ошибки [23, 24]. Он позволяет учесть ошибку между текущим выходом нейросети и требуемым значением в каждом весовом коэффициенте ИНС. Данный подход основан на методе градиентного спуска. Ошибку "распространяют" (с помощью вычисления градиента функции потерь (Loss Function L)) от выхода к входу ИНС. Функция L учитывает ошибку каждого выхода сети по сравнению эталонным. Исходя из накопленного опыта для формирования набора входных сигналов (обучающей выборки) при синтезе нейроимитатора и нейрорегулятора для объектов с ограничителями предлагается руководствоваться следующими рекомендациями.

1. Сигналы должны отражать особенности работы объекта как в переходном, так и в установившемся режимах. В общем случае следует использовать типовые непрерывные и разрывные сигналы (синус и меандр разных амплитуд и частот). Меандр позволяет получить информацию о реакции объекта на скачкообразное изменение и постоянное значение входа. Гармоники используются для сбора данных об обработке непрерывных, динамически изменяющихся сигналов. Обучающие последовательности следует разделить на две категории: ограничители в ОУ не достигаются; ограничители достигаются, причем если ограничителей в объекте несколько, то достигаются каждый в отдельности и вместе. При этом желательно, чтобы в обучающей выборке число данных в категориях было примерно одинаковым.

2. На все выбранные сигналы необходимо записать реакцию объекта управления. Собранные данные следует подготовить. Для сетей прямого распространения записанные сигналы разбиваются на пары так, чтобы они отображали зависимость входных данных и результата на одном такте квантования, т. е. для нейроимитатора (nn_3): $x_i = nn_3(u_i, x_{i-1}, x_{i-2}, \dots)$, а для

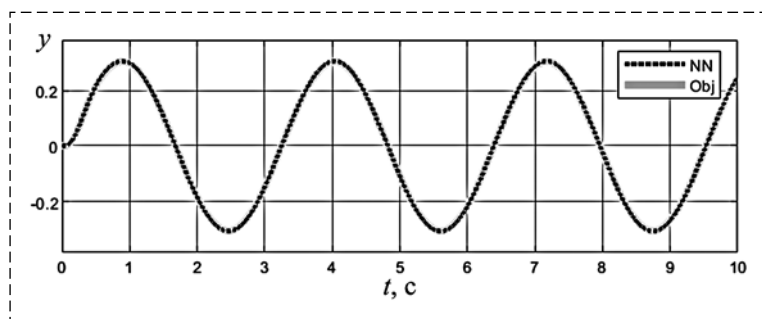


Рис. 6. Выход объекта и имитатора без достижения ограничителей
Fig. 6. Object and simulator output without reaching the limiters

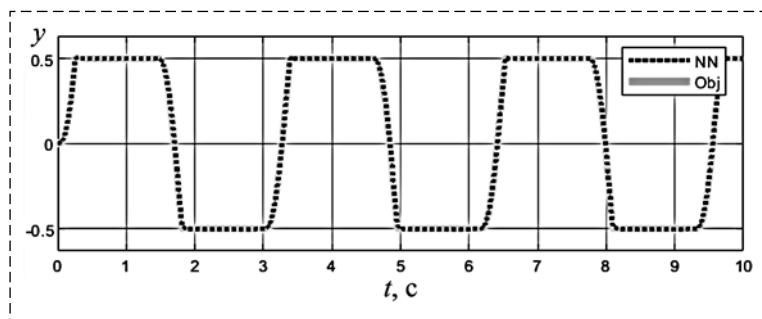


Рис. 7. Выход объекта и имитатора с достижением ограничителей
Fig. 7. Output of the object and simulator with reaching the limiters

нейрорегулятора (nn_p): $u_i = nn_p(r_i, x_{i-1}, x_{i-2}, \dots)$, где i — номер дискретного шага, u — сигнал управления объектом, x — состояние объекта, r — полезный входной сигнал (см. рис. 1). Таким образом, собранные пары никак не зависят друг от друга и от того, из какого сигнала они были получены, что позволяет ускорить обучение за счет параллельных вычислений.

Синтез нейрорегулятора на основе обратного распространения ошибки через нейроимитатор

На основании изложенного выше предлагается следующий алгоритм синтеза нейрорегулятора для объектов с ограничителями.

1. Выбирается структура нейрорегулятора, т. е. определяются типы слоев, число нейронов в каждом слое, функции активации согласно сформулированным рекомендациям.

2. Формируется обучающая выборка данных, отражающая динамику работы ОУ, в соответствии с приведенными выше соображениями. Полученную тренировочную выборку следует разбить на две части: тренировочную и проверочную. Обычно выборку делят в пропорции 90 % к 10 % соответственно [25].

3. Проводится обучение нейроимитатора методом обратного распространения ошибки. Универсальным способом является разделение данных на мини-пакеты (mini-batch) с использованием вариаций адаптивного стохастического градиентного спуска, например алгоритма Adam [26].

В качестве функции потерь обычно используется среднеквадратичная ошибка (MSE-loss)

$$\text{loss}(\mathbf{a}, \tilde{\mathbf{a}}) = \frac{1}{k} \sum_{i=1}^k (\mathbf{a}_i - \tilde{\mathbf{a}}_i), \quad (8)$$

где \mathbf{a} и $\tilde{\mathbf{a}}$ — векторы с входными (полученными в результате расчета ИНС на данных из обучающей выборки) и целевыми значениями соответственно; k — число значений в обучающей выборке.

Следует повторять эпохи обучения до тех пор, пока функция (8) не достигнет целевого

значения, которое определяется разработчиком нейрорегулятора.

4. Определяется структура нейрорегулятора и выполняется его обучение методом обратного распространения ошибки. На каждом такте квантования вычисляется значение управления на выходе нейрорегулятора на основе задающего сигнала для системы и обратных связей от объекта. Результатом вычислений является значение управляющего сигнала, на основе которого нейроимитатором определяется новое состояние ОУ. Затем полученное рассогласование (8) методом обратного распространения ошибки передают через нейроимитатор и нейрорегулятор. При этом обучается только нейрорегулятор.

5. Проводится оценка качества полученной системы в различных режимах работы. Так как обучение происходит в реальном времени, то в случае ухудшения качества управления (например, при изменении параметров объекта управления) возможно выполнить дополнительное обучение как нейроимитатора, так и нейрорегулятора. Таким образом система получает свойство адаптивности, которое, однако, может быть использовано только для медленно изменяющихся параметров. Это обусловлено тем, что процесс обучения требует больших вычислительных ресурсов.

Пример. На основе разработанных алгоритмов был синтезирован нейросетевой регулятор для следящего гидропривода, упрощенная функциональная модель которого представлена на рис. 8.

На рис. 8: γ — угол поворота люльки гидронасоса; p — перепад давления в гидромоторе; ω — скорость вращения выходного вала; φ — угол поворота вала. В структурной схеме учитываются присущие гидроприводу ограничитель на угол поворота люльки насоса: $|\gamma| \leq D_1$ и безынерционный ограничитель перепада давления масла в силовых гидромагистралях: $|p| \leq D_2$.

Точная математическая модель для построения нейрорегулятора в общем случае не требуется, но должно быть сделано предположение о ее порядке.

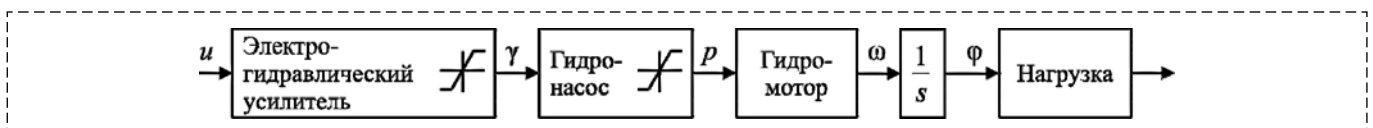


Рис. 8. Структурно-функциональная схема гидропривода

Fig. 8. Structure-functional diagram of the hydraulic drive

На основе разработанных алгоритмов были выбраны следующие структуры нейросетей. Если считать, что ОУ описывается системой нелинейных дифференциальных уравнений четвертого порядка, то для нейроимитатора минимально необходимо пять нейронов на входе, четыре нейрона без функций активации в первом скрытом слое, во втором слое 12 нейронов с функцией активации ReLU и с весами смещения, в выходном слое четыре нейрона. Для нейрорегулятора выбрана сеть прямого распространения с четырьмя входами (вектор состояния), двумя скрытыми слоями: в первом — 24 нейрона, во втором — восемь нейронов с функциями активации leakyReLU и на выходе один нейрон, формирующий сигнал на объект управления.

Обучение нейроимитатора и нейрорегулятора проводили согласно предложенным выше алгоритмам. Для оценки полученных результатов был проведен сравнительный анализ синтезированного нейрорегулятора с квазиоптимальным законом управления, сформированным в работе [27]. На рис. 9 (см. вторую сторону обложки) показаны ошибки слежения e за эквивалентным гармоническим сигналом, где штриховой линией обозначен выход нейросетевой системы, а сплошной линией — выход системы с квазиоптимальным регулятором по быстродействию. Из графиков видно, что данные замкнутые системы имеют близкие амплитуды ошибки слежения за входным сигналом управления (0,016 рад у системы с квазиоптимальным регулятором по быстродействию и 0,018 рад — у системы с нейрорегулятором).

Заключение

Результаты работы показывают, что в некоторых случаях использование ИНС в замкнутых следящих системах автоматического управления позволяет упростить и формализовать задачу синтеза регулятора. Выполнен анализ существующих типов ИНС и известных методов синтеза нейросетевых систем управления. Для нейросетевых регуляторов при наличии ограничителей в объекте управления были выбраны нейронные сети прямого распространения, а для их обучения — метод обратного распространения ошибки через нейроимитатор. В качестве функций активации рекомендовано использовать ReLU или leakyReLU.

Разработан алгоритм формирования структуры нейрорегулятора и нейроимитатора для объекта управления с ограничителями, позволяющий определить минимальное число слоев и нейронов в них. При синтезе нейрорегулятора четко определены размеры входного и выходного слоев. Для скрытых слоев следует пользоваться правилом поэтапного увеличения нейронов и числа слоев до достижения результата, соответствующего заданным требованиям. Разработан алгоритм формирования обучающей выборки с учетом наличия в объекте управления ограничителей. Предложен метод синтеза нейрорегулятора, позволяющий определить основные параметры используемых нейронных сетей и алгоритмы обучения. Совокупность полученных результатов позволяет сформировать методику синтеза и оптимизации нейросетевых систем управления для управления динамическими объектами с ограничителями в условиях неполной информации. Применение указанной методики продемонстрировано на примере конкретного технического объекта.

Список литературы

1. Sünderhauf N., Brock O., Scheirer W., Hadsell R., Fox D., Leitner J., Upcroft B., Abbeel P., Burgard W., Milford M., Corke P. The Limits and Potentials of Deep Learning for Robotics // arXiv:1804.06557. arXiv. 2018.
2. Beard R. W., McLain T. W. Small Unmanned Aircraft: Theory and Practice. Small Unmanned Aircraft. Princeton University Press, 2012. P. 317.
3. Kaess M., Johannsson H., Roberts R., Ila V., Leonard J. J., Dellaert F. iSAM2: Incremental smoothing and mapping using the Bayes tree // The International Journal of Robotics Research. 2012. Vol. 31, N. 2. P. 216—235.
4. Huang S., Papernot N., Goodfellow I., Duan Y., Abbeel P. Adversarial Attacks on Neural Network Policies. arXiv:1702.02284. arXiv. 2017.
5. Bendale A., Boulton T. Towards Open World Recognition // Proceeding of 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2015. P. 1893—1902.
6. Fabiani F., Goulart P. J. Neural network controllers for uncertain linear systems // arXiv:2204.13209. arXiv. 2022.
7. Jabali M. B. A., Kazemi M. H. Uncertain polytopic LPV modelling of robot manipulators and trajectory tracking // International Journal of Control, Automation and Systems. 2017. Vol. 15, N. 2. P. 883—891.
8. Li S., Ozturk E., De Wagter C., Croon G. C. H.E. de, Izzo D. Aggressive Online Control of a Quadrotor via Deep Network Representations of Optimality Principles // Proceeding of 2020 IEEE International Conference on Robotics and Automation (ICRA), Paris, France: IEEE, 2020. P. 6282—6287.
9. Чернодуб А. Н., Дзюба Д. А. Обзор методов нейрорегуляторов // Проблемы программирования. 2011. № 2. С. 79—94.
10. Ekman M. Learning Deep Learning: Theory and Practice of Neural Networks, Computer Vision, Natural Language Processing, and Transformers Using TensorFlow. Learning Deep Learning. Addison-Wesley, 2022. 688 P.

11. **Izzo D., Öztürk E.** Real-Time Guidance for Low-Thrust Transfers Using Deep Neural Networks // Journal of Guidance, Control, and Dynamics. 2021. Vol. 44, N. 2. P. 315—327.
12. **Reifman J., Feldman E. E.** Nonlinear programming with feedforward neural networks // IJCNN'99. International Joint Conference on Neural Networks. Proceedings (Cat. No. 99CH36339). 1999. Vol. 1. P. 594—598.
13. **Effati S., Baymani M.** A new nonlinear neural network for solving quadratic programming problems // Applied Mathematics and Computation. 2005. Vol. 165, N. 3. P. 719—729.
14. **Niestroy M.** Optimal feedback controller approximation using neural networks and nonlinear programming techniques // 1998 IEEE International Joint Conference on Neural Networks Proceedings. IEEE World Congress on Computational Intelligence (Cat. No.98CH36227). 1998. Vol. 3. P. 2110—2115.
15. **Nakamura-Zimmerer T., Gong Q., Kang W.** A Causality-Free Neural Network Method for High-Dimensional Hamilton-Jacobi-Bellman Equations // Proceeding of 2020 American Control Conference (ACC). 2020. P. 787—793.
16. **Nakamura-Zimmerer T., Gong Q., Kang W.** Adaptive Deep Learning for High-Dimensional Hamilton-Jacobi-Bellman Equations // SIAM Journal on Scientific Computing. 2021. Vol. 43, N. 2. P. A1221—A1247.
17. **Nakamura-Zimmerer T., Gong Q., Kang W.** QRnet: Optimal Regulator Design With LQR-Augmented Neural Networks // IEEE Control Systems Letters. 2021. Vol. 5, N. 4. P. 1303—1308.
18. **Yang X., Liu D., Huang Y.** Neural-network-based online optimal control for uncertain non-linear continuous-time systems with control constraints // IET Control Theory & Applications. 2013. Vol. 7, N. 17. P. 2037—2047.
19. **Parrilo P. A.** Structured Semidefinite Programs and Semi-algebraic Geometry Methods in Robustness and Optimization. California Institute of Technology, 2000. P. 234.
20. **Lofberg J.** Pre- and Post-Processing Sum-of-Squares Programs in Practice // IEEE Transactions on Automatic Control. 2009. Vol. 54, N. 5. P. 1007—1011.
21. **Feofilov S. V., Khapkin D. L.** Synthesis of neural network controllers for objects with non-linearity of the constraint type // Journal of Physics: Conference Series. 2021. Vol. 1958, N. 1. P. 012014.
22. **Nair V., Hinton G. E.** Rectified Linear Units Improve Restricted Boltzmann Machines // Proceedings of the 27th International Conference on International Conference on Machine Learning ICML'10, Madison, WI, USA, Omnipress, 2010, P. 807—814.
23. **Галушкин А. И.** Синтез многослойных систем распознавания образов. М.: Энергия, 1974. 367 с.
24. **Werbos P. J.** Backpropagation through time: what it does and how to do it // Proceedings of the IEEE. 1990. Vol. 78, N. 10. P. 1550—1560.
25. **Goodfellow I., Bengio Y., Courville A.** Deep learning: Adaptive computation and machine learning. Cambridge, Massachusetts: The MIT Press, 2016. P. 775.
26. **Kingma D. P., Ba J.** Adam: A Method for Stochastic Optimization // Adam. arXiv:1412.6980. arXiv. 2017.
27. **Хоанг Ч. К.** Оптимизация объемных силовых следящих гидроприводов по быстродействию и по точности режима слежения / диссертация... кандидата технических наук: 05.13.01. Тула, 2006. 131 с.

Structural and Parametric Synthesis of Neural Network Controllers for Control Objects with Limiters

S. V. Feofilov, svfeofilov@mail.ru, **A. V. Kozyr**, kozyr_a_v@mail.ru, **D. L. Khapkin**, dima-hapkin@ya.ru, Tula State University, Tula, 300012, Russian Federation

Corresponding author: Feofilov Sergey V., Dr. of Eng. Sc., Professor, Tula State University, Tula, 300012, Russian Federation, e-mail: svfeofilov@mail.ru

Accepted on August 17, 2023

Abstract

The article presents a methodology for the synthesis of digital control systems for nonlinear objects with limiters under conditions of incomplete information. Closed-loop tracking systems with negative feedback are considered. Artificial neural networks are proposed to build a controller, which is included in series with the control object. This approach is effective when known classical methods do not allow to synthesize control. This is the case, for example, if the mathematical model is essentially nonlinear and is not fully defined. The developed methods allow us to expand the class of technical systems, for which the direct (without using various kinds of simplifications) synthesis of control laws that are close to optimal is possible. In addition, neural network controllers possess the properties of robustness, adaptivity, and are initially digital, i.e. those qualities, which are very much in demand in practice. In article main attention is given to such problems, as a choice of neural network structure for neural simulator and neural controller, construction of training sample, ensuring convergence of the process of weights correction. For training neural networks the method of back propagation of error is used as a basic one. The effectiveness of the proposed technique is demonstrated by the example of the synthesis of a neuroregulator for a specific technical object and its comparison with classical control systems. It should be noted that today neural network technologies are widespread enough in various spheres of activity. The successes demonstrated in sound processing, image processing, automatic translation, in navigation systems, in big data processing are impressive. However, their application in automatic control systems is not so widespread. The authors of this article believe that the potential of artificial neural networks can be used in this direction. It should be understood that the use of neural networks is effective only under certain conditions and properties of the control object.

Keywords: neural network controller, tracking system, learning sampling, neurosimulator

Acknowledgements: The research was carried out at the expense of the grant of the Russian Science Foundation No. 23-29-00609.

For citation:

Feofilov S. V., Kozyr A. V., Khapkin D. L. Structural and Parametric Synthesis of Neural Network Controllers for Control Objects with Limiters, *Mekhatronika, Avtomatizatsiya, Upravlenie*, 2023, vol. 24, no. 11, pp. 563—572.

DOI: 10.17587/mau.24.563-572

References

1. **Sünderhauf N., Brock O., Scheirer W., Hadsell R., Fox D., Leitner J., Upcroft B., Abbeel P., Burgard W., Milford M., Corke P.** The Limits and Potentials of Deep Learning for Robotics, *arXiv:1804.06557, arXiv*, 2018.
2. **Beard R. W., McLain T. W.** Small Unmanned Aircraft: Theory and Practice. Small Unmanned Aircraft, Princeton University Press, 2012, pp. 216–235.
3. **Kaess M., Johannsson H., Roberts R., Ila V., Leonard J. J., Dellaert F.** iSAM2: Incremental smoothing and mapping using the Bayes tree, *The International Journal of Robotics Research*, 2012, vol. 31, no. 2, pp. 216–235.
4. **Huang S., Papernot N., Goodfellow I., Duan Y., Abbeel P.** Adversarial Attacks on Neural Network Policies, *arXiv:1702.02284, arXiv*, 2017.
5. **Bendale A., Boulton T.** Towards Open World Recognition, *Proceeding of 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015, pp. 1893–1902.
6. **Fabiani F., Goulart P. J.** Neural network controllers for uncertain linear systems, *arXiv:2204.13209, arXiv*, 2022.
7. **Jabali M. B. A., Kazemi M. H.** Uncertain polytopic LPV modelling of robot manipulators and trajectory tracking, *International Journal of Control, Automation and Systems*, 2017, vol. 15, no. 2, pp. 883–891.
8. **Li S., Ozturk E., De Wagter C., Croon G. C. H. E. de, Izzo D.** Aggressive Online Control of a Quadrotor via Deep Network Representations of Optimality Principles, *Proceeding of 2020 IEEE International Conference on Robotics and Automation (ICRA)*, Paris, France, IEEE, 2020, pp. 6282–6287.
9. **Chernodub A. N., Dzyuba D. A.** Review of methods, Problemy programirovaniya, 2011, no. 2, pp. 79–94 (In Russian).
10. **Ekman M.** Learning Deep Learning: Theory and Practice of Neural Networks, Computer Vision, Natural Language Processing, and Transformers Using TensorFlow. Learning Deep Learning, Addison-Wesley, 2022, 688 p.
11. **Izzo D., Öztürk E.** Real-Time Guidance for Low-Thrust Transfers Using Deep Neural Networks, *Journal of Guidance, Control, and Dynamics*, 2021, vol. 44, no. 2, pp. 315–327.
12. **Reifman J., Feldman E. E.** Nonlinear programming with feedforward neural networks, *IJCNN'99. International Joint Conference on Neural Networks. Proceedings (Cat. No.99CH36339)*, 1999, vol. 1, pp. 594–598.
13. **Effati S., Baymani M.** A new nonlinear neural network for solving quadratic programming problems, *Applied Mathematics and Computation*, 2005, vol. 165, no. 3, pp. 719–729.
14. **Niestroy M.** Optimal feedback controller approximation using neural networks and nonlinear programming techniques, *1998 IEEE International Joint Conference on Neural Networks Proceedings. IEEE World Congress on Computational Intelligence (Cat. No.98CH36227)*, 1998, vol. 3, pp. 2110–2115.
15. **Nakamura-Zimmerer T., Gong Q., Kang W.** A Causality-Free Neural Network Method for High-Dimensional Hamilton-Jacobi-Bellman Equations, *Proceeding of 2020 American Control Conference (ACC)*, 2020, pp. 787–793.
16. **Nakamura-Zimmerer T., Gong Q., Kang W.** Adaptive Deep Learning for High-Dimensional Hamilton--Jacobi--Bellman Equations, *SIAM Journal on Scientific Computing*, 2021, vol. 43, no. 2, pp. A1221–A1247.
17. **Nakamura-Zimmerer T., Gong Q., Kang W.** QRnet: Optimal Regulator Design With LQR-Augmented Neural Networks, *IEEE Control Systems Letters*, 2021, vol. 5, no. 4, pp. 1303–1308.
18. **Yang X., Liu D., Huang Y.** Neural-network-based online optimal control for uncertain non-linear continuous-time systems with control constraints, *IET Control Theory & Applications*, 2013, vol. 7, no. 17, pp. 2037–2047.
19. **Parrilo P. A.** Structured Semidefinite Programs and Semi-algebraic Geometry Methods in Robustness and Optimization, *California Institute of Technology*, 2000, pp. 234.
20. **Lofberg J.** Pre- and Post-Processing Sum-of-Squares Programs in Practice, *IEEE Transactions on Automatic Control*, 2009, vol. 54, no. 5, pp. 1007–1011.
21. **Feofilov S. V., Khapkin D. L.** Synthesis of neural network controllers for objects with non-linearity of the constraint type, *Journal of Physics: Conference Series*, 2021, vol. 1958, no. 1, pp. 012014.
22. **Nair V., Hinton G. E.** Rectified Linear Units Improve Restricted Boltzmann Machines, *Proceedings of the 27th International Conference on International Conference on Machine Learning ICML'10*, Madison, WI, USA, Omnipress, 2010, pp. 807–814.
23. **Galushkin A. I.** Synthesis of multilayer image recognition systems, Moscow, Energiya, 1974, 367 p. (In Russian).
24. **Werbos P. J.** Backpropagation through time: what it does and how to do it, *Proceedings of the IEEE*, 1990, vol. 78, no. 10, pp. 1550–1560.
25. **Goodfellow I., Bengio Y., Courville A.** Deep learning: Adaptive computation and machine learning, Cambridge, Massachusetts, The MIT Press, 2016, 775 p.
26. **Kingma D. P., Ba J. A.** A Method for Stochastic Optimization, *arXiv:1412.6980, arXiv*, 2017.
27. **Hoang C. K.** Optimization of volumetric power tracking hydraulic drives in terms of speed and accuracy of the tracking mode, Ph.D dissertation (05.13.01), Tula, 2006, 131 p. (In Russian).