

АДАПТИВНАЯ СИСТЕМА ТРАЕКТОРНОГО УПРАВЛЕНИЯ АНПА НА БАЗЕ НЕЙРОННОЙ СЕТИ ПРЯМОГО РАСПРОСТРАНЕНИЯ

Романова В. Р.¹

*(Крымский федеральный университет
имени В.И. Вернадского, Симферополь)*

Зуев С. В.²

*(Белгородский государственный технологический
университет им. В.Г. Шухова, Белгород)*

Работа посвящена разработке и исследованию интеллектуальной системы адаптивного автоматического управления с заданной целью, основанной на использовании искусственной нейронной сети прямого распространения. Объектом управления является автономный необитаемый подводный аппарат (АНПА). В работе предлагается подать на вход нейронной сети сигналы, получаемые от систем АНПА, и использовать выходной сигнал нейронной сети для управления с целью удержания аппарата на заданной траектории. В качестве результата работы предлагается модель и метод обучения, которые приводят к удержанию АНПА на заданной траектории при внешнем воздействии с естественным ограничением для рассматриваемого мобильного робота. При наличии непрерывной заданной траектории и дискретных сигналов от систем АНПА это позволяет следовать заданной траектории с помощью простой интеллектуальной системы управления, не требующей больших вычислительных мощностей. Предложенный метод обучения системы управления допускает предварительное обучение на численной модели движения аппарата со случайными внешними воздействиями, но не требует такого предварительного обучения при определенных условиях. Показано, в частности, что в случае достаточно большой скорости обучения модель успевает перестраиваться и реагирует на изменившиеся обстоятельства. Предложенная интеллектуальная система адаптивного автоматического управления может найти применение в тех случаях, когда характерное время изменений в системе имеет порядок времени обучения, а траектория движения удовлетворяет изложенным в работе требованиям.

Ключевые слова: искусственная нейронная сеть, автономный обитаемый подводный аппарат, система адаптивного управления, интеллектуальная система управления, обучение искусственной нейронной сети.

¹ Валерия Романовна Романова, студент (lero4ka2004ro@gmail.com).

² Сергей Валентинович Зуев, к.ф.-м.н., доцент (sergey.zuev@bk.ru).

1. Введение

Задача адаптивного управления является одной из наиболее важных в области автоматического управления. Она заключается в разработке системы управления, способной эффективно адаптироваться к изменениям внешних условий.

Адаптивное управление используется там, где происходит изменение внешних условий, влияющих на работу системы, – это может происходить практически в любом виде деятельности. Внешние условия меняются независимо от состояния рабочей системы и влияют на ее работу по-разному, что с необходимостью приводит к учету в системах управления такого воздействия: предусматривается адаптивное управление, которое компенсирует внешние изменения за счет дополнительного управляющего воздействия. Задача адаптивного управления была поставлена в середине XX века [3], но ее универсальное решение пока отсутствует. Использование для этих целей систем искусственного интеллекта считается перспективным [31].

В этой работе рассматривается адаптивное управление автономным необитаемым подводным аппаратом (АНПА) с помощью искусственной нейронной сети (далее – ИНС) прямого распространения. Такой выбор (АНПА – ИНС прямого распространения) связан с тем, для АНПА характерное время изменений в системе, равное отношению размера аппарата к его скорости, больше, чем у большинства других мобильных роботов. Поэтому задача для такой системы управления допускает более сложные вычисления. То есть предложенная идея адаптивного управления может использоваться и в других системах, в которых имеется достаточно времени или ресурсов для вычислений, но из наиболее распространенных сейчас систем для этой цели хорошо подходит именно АНПА.

Автономные необитаемые подводные аппараты – это подводные мобильные роботы, которые предназначены для получения данных о внешней среде или воздействия на нее в автоматическом режиме, зачастую без информационно-командной связи с человеком-оператором. Применение АНПА является широким и включает в себя (не ограничиваясь этим) обследования мест авиакатастроф, океанографические исследования, разнообраз-

ные исследования морского дна, доставку грузов, производственные операции, добычу полезных ископаемых (например, аппараты компании Impossible Metals, <https://impossiblemetals.com>). Диапазон инновационных применений АНПА также огромен [19, 33].

Указанные задачи часто невозможно выполнить без использования автоматизированных систем управления АНПА. За последние годы было предложено множество различных схем управления. Проблема отслеживания траектории исследовалась в работе [15] с помощью модели сингулярных возмущений. Модельный прогностический контроллер для АНПА был предложен в [23]. Система локализации и планирования маршрута с учетом рельефа для навигации в течение длительного периода времени и определения местоположения АНПА описана в работе [27]. В работе [6] было разработано отказоустойчивое отслеживающее устройство. Координированное управление системой нескольких АНПА рассматривалось в [29].

Отдельно стоит указать системы управления, предложенные в связи с ограничениями вычислительной производительности бортовых вычислительных систем АНПА. Адаптивный не-сингулярный контроллер скользящего режима для АНПА, проявляющих только ответную активность (в норме пассивных), исследовался в работе [24]. В статье [22] модель АНПА была создана с использованием метода Лагранжа, а управление движением было организовано с помощью адаптивного контроллера скользящего режима на основе кватернионов. Ограниченный прогностический контроллер для АНПА был предложен в [21]. Контроллер на нечеткой логике рассмотрен в [12]. Оптимальный робастный контроллер для задачи отслеживания траектории АНПА исследован в [32].

В этой работе имеет существенное значение дискретность получения сигналов, используемых для управления. Поэтому имеет смысл указать и работы, направленные на преодоление ограничений управления в условиях задержек связи. Например, алгоритм управления формированием сигнала был разработан в [15], и был предложен контроллер, основанный на наблюдении расширенного состояния.

Таким образом, в данных работах рассмотрены различные подходы к адаптивному управлению в контексте разных задач управления и ситуаций, использованы различные подходы. Однако интеллектуальные системы управления использовались только в работах [12, 21], т.е. адаптивность к широкому спектру воздействий рассматривалась только в них. Вопрос об адаптивности к произвольным воздействиям по-прежнему открыт и интеллектуальная система на основе ИНС является перспективным кандидатом на его решение. Преодоление ограничений адаптации к произвольным внешним воздействиям особенно ценно при учете малых доступных вычислительных мощностей и задержек в получении данных.

Фактически АНПА представляют собой весьма неопределенные, связанные, изменяющиеся во времени и нелинейные системы [10]. Большинство моделей АНПА чрезвычайно упрощены из-за неточного характера методов и сложности моделирования, а также неизвестных корреляций и взаимосвязей между переменными, что приводит к несовершенству контроллеров для реальных подводных миссий. Немоделируемая динамика является неотъемлемой частью АНПА, впрочем, как и беспилотных летательных аппаратов (БПЛА), но характер управления для этих двух классов мобильных роботов все же существенно различен.

Интеграция искусственной нейронной сети в систему управления АНПА может быть сделана на основе мультиагентной архитектуры системы управления, описанной в [1]. На рис. 1 представлена схема системы управления из [1], в которой ИНС может быть интегрирована в сеть обмена данными в качестве посредника между системой навигации, системами позиционирования с одной стороны, и движительно-рулевым комплексом и системой дифферентовки – с другой стороны.

Обычно считается, что задача оптимального управления нелинейными системами решается как уравнение Гамильтона – Якоби – Беллмана, которое является очень сложным и не может быть решено напрямую для реальных систем. Реальные системы могут быть описаны как сложные, в данном случае – динамически сложные, т.е. их поведение не является детерминированным и, строго говоря, не определяется решением какого-либо урав-

нения. Моделирование таких систем часто делается с помощью машинного обучения. Этот метод использует закономерности, которые есть в системе, не предьявляя их явно.



Рис. 1. Схема системы управления АНПА [17]

Одним из самых распространенных моделей машинного обучения является искусственная нейронная сеть (ИНС). Среди методов обучения ИНС есть обучение с подкреплением [28], которое является подходящим в случае изменчивости среды. Обучение с подкреплением подразумевает обратную связь от системных переменных и постоянно улучшает предсказание работы системы [4, 5, 11]. Для оценки текущей эффективности системы используется обратная связь от управляющего воздействия и последующие шаги управления делаются с учетом этой обратной связи и ее влияния на достижение цели.

Применительно к настоящей работе можно указать следующие современные исследования в области интеллектуального управления с обучением с подкреплением. В [25] был разработан контроллер отслеживания траектории АНПА с помощью обучения с подкреплением. Такой же вид обучения был использован в работе [34] для решения задачи оптимального управления для систем с насыщением и отказами. Проблема адаптивной стратегии обучения с подкреплением была изучена для класса

аффинных нелинейных систем с дискретным временем в работе [13]. Для класса стохастических нелинейных систем в [30] был разработан упрощенный оптимизированный метод управления с использованием обучения с подкреплением. В [20] разработали оптимальный контроллер отслеживания для мультиагентных систем с дискретным временем. Контроллер, основанный на обучении с подкреплением, был предложен в [35] для малоактивных (в норме пассивных) морских надводных судов. Оптимальный контроллер оценки бесконечного горизонта был предложен для системы с неизвестными параметрами дрейфа в [8].

Несмотря на указанные достижения в области контроллеров на основе обучения с подкреплением для нелинейных систем, практически везде имело место насыщение привода. Насыщение привода происходит, когда контроллер генерирует уровень управляющего сигнала, превышающий допустимые пределы работы приводов. Такая ситуация наносит вред как контроллеру, так и транспортному средству и, кроме того, приводит к сбою управления. Чтобы решить эту проблему, необходимо разработать системы управления с защитой от насыщения, что является математически сложной задачей [9]. Одним из путей ее решения является компенсация нелинейности насыщения привода с помощью нейронных сетей.

В нашей работе делается попытка построения простого контроллера, претендующего на преодоление указанных выше проблем накопления ошибок и насыщения с минимальными требованиями к вычислительным мощностям.

2. Постановка задачи

Начальное положение аппарата $p^k(0)$, $k = 1, 2, 3$, и его желаемую трехмерную траекторию движения $x^k(t)$, $t \in [0, T]$, будем считать известными. На основе дополнительных предположений о нахождении внешних воздействий в определенном диапазоне в течение всего времени движения от 0 до T требуется обеспечить движение АНПА по траектории с минимальными отклонениями, используя данные бортовой инерциальной навигационной системы, данные системы позиционирования, обновляющи-

еся с определенными частотами, а также произведенные управляющие воздействия.

Будем считать, что имеется некоторая система позиционирования АНПА, использующая как инерциальные данные, так и возможные данные от реперных точек на дне, которая после первоначального определения положения АНПА периодически отслеживает изменение позиции аппарата и выдает его текущее положение в той же системе координат, в которой задано начальное положение и желаемая траектория АНПА. Таким образом, имеются значения $p^k(t_j)$ в некоторые моменты времени t_j .

Для дальнейшего требуется описать ограничения на допустимые внешние воздействия. Под ними будем подразумевать скорость потока воды в месте нахождения аппарата в момент времени t . Обозначим компоненты этой скорости через $f^k(t)$. Тогда корректная работа контроллера будет возможна только в случае

$$\int_0^T u(t)dt > \int_0^T f(t)dt,$$

где $u(t)$, $f(t)$ – абсолютные моментальные значения скоростей потоков в связанной системе координат и в абсолютной системе координат соответственно. Физически это условие означает, что управляющее воздействие в течение времени T превышает случайные внешние воздействия за то же время. На самом деле для корректной работы контроллера необходимо, чтобы разница в этом неравенстве была хотя бы на порядок величины.

Как уже упоминалось, выбор АНПА в качестве объекта управления связан с тем, что у этого вида мобильных роботов сравнительно большое характерное время. Например, для аппарата длиной 2 м, движущегося с максимальной скоростью 10 м/с, это время равно 200 мс. Процесс обучения ИНС рассматриваемого далее вида занимает около 30 мс, если имеются все данные. Данные, поступающие с наибольшими задержками, – это обычно позиционирование. Будем считать, что даже эти данные не задерживают процесс обучения на время, более чем на порядок превышающее характерное время для объекта. Тогда можно считать, что все изменения за цикл обучения относительно малы.

Управляющее воздействие на АНПА в нашей модели производится одинаковыми двигателями, число которых равно S . Ничто не мешает включить в управляющие элементы еще и систему дифферентовки, но тогда нужно будет описывать преобразования ее сигналов отдельно. Это лишь загромоздит рассмотрение без определенного смысла. Поэтому далее рассматриваем только двигатели. Каждый из них может воспринимать управляющий сигнал m^s , причем минимальный промежуток времени между последовательными командами равен θ_m . Конечно, результат воздействия на режим движения АНПА разных управляющих сигналов зависит нелинейным образом от самого сигнала и от внешних факторов (f), но мы будем считать, что эта зависимость в результате обучения будет содержаться в структуре данных весов и смещений ИНС, поскольку при обучении используется результирующее положение АНПА.

Таким образом, в задаче определены следующие сигналы:

1) $u^l(t)$ – ускорения и угловые скорости АНПА; в каждый момент времени это шестикомпонентный вектор, который предоставляется инерциальной навигационной системой; эти сигналы поступают с периодичностью θ_u и имеют квант η (одинаковый для всех компонент);

2) $p^k(t)$ – позиционирование АНПА, поступающее из подсистемы позиционирования; эти сигналы поступают в систему с периодичностью θ_p и имеют квант ξ (погрешность подсистемы положим постоянной и одинаковой для всех осей);

3) $f^k(t)$ – компоненты скорости внешнего течения (в абсолютной системе координат), не измеряемые и непредсказуемые; не имеет периодичности и установленного кванта;

4) $m^s(t)$ – управляющее воздействие – сигнал, который система подает двигателям, связанный какой-то функцией с их тягой (вид функции не имеет значения); в момент времени t имеются S значений $m^s(t)$, а работа управляющей системы состоит в том, чтобы получить столько же значений $m^s(t + \theta_m)$ и применить их в момент $t + \theta_m$; квантование этих сигналов задается возможностями управления и пусть квант равен μ ;

5) $x^k(t)$ – желаемая траектория; задается последовательностью значений с дискретностью θ_k , которую мы можем установить произвольно.

В этом случае решением задачи будет такая (зависящая от $p^k(t)$) последовательность векторов $m^s(t)$, которая приведет к минимальному отклонению $p^k(t)$ от $x^k(t)$ при любом допустимом (в вышеуказанном смысле) внешнем воздействии.

3. Метод решения с помощью алгоритма прямого распространения

Искусственная нейронная сеть – это компьютерная система, реализующая концепцию машинного обучения, т.е. построения модели процесса на примерах его работы. О нейронных сетях написано огромное число работ и нет смысла делать их полный обзор, но используемые в данном исследовании ИНС прямого распространения хорошо описаны, например, в книгах [2] и [7].

ИНС прямого распространения (FFNN) – это простой тип нейронной сети, в котором информация передается только в одном направлении: от входных нейронов к выходным. Такие ИНС просты в реализации и сравнительно быстро обучаются, имеют хорошую производительность на задачах классификации и регрессии [14, 17, 18]. Недостатками таких ИНС является плохая адаптация к обработке последовательностей или данных, меняющихся со временем. Поэтому, казалось бы, такая архитектура ИНС не очень подходит для поставленной в нашей работе задачи.

Однако если переформулировать задачу так, чтобы в каждый фиксированный сравнительно небольшой промежуток времени система управления обеспечивала уменьшение отклонения аппарата от траектории, то такая задача уже не является обусловленной прошлыми значениями сигналов и может быть успешно решена ИНС прямого распространения. Иными словами, мы предлагаем заново обучать систему сразу после того как получили прямой проход на только что полученных весах и смещениях. С учетом того, что характерное время объекта управления того же порядка величины, что и время обучения и работы ИНС, таким образом получится онлайн обучаемая си-

стема, постоянно воспринимающая новые данные. Мало того, такая система будет реализовывать концепцию обучения с подкреплением, так как регулярно получаемая оценка имеется – это разность выдачи ИНС в канале позиционирования с данными траектории.

В задаче имеется несколько величин, характеризующих временную дискретность: θ_u , θ_p , θ_m , θ_x , причем θ_x можно выбирать произвольно: непрерывную кривую, соединяющую начальное и конечное положение аппарата, можно представить последовательностью с любой дискретностью. Остальные величины зависят от характеристик инерциальной навигационной системы, возможностей системы позиционирования и вычислительных мощностей системы управления – эти величины нельзя менять произвольно.

Для расчетов на ЭВМ все величины так или иначе представляются целыми числами. То есть можно выбрать такую величину кванта времени τ , что все указанные выше величины дискретностей будут целыми положительными числами.

Можно выбрать τ настолько малым, что $T_x = \theta_x / \tau$ будет с достаточной степенью точности наибольшим общим делителем величин $T_u = \theta_u / \tau$, $T_p = \theta_p / \tau$, $T_m = \theta_m / \tau$. Поэтому далее будем использовать именно целочисленную нотацию, предполагая справедливость следующих соотношений:

$$(1) \quad T_u, T_p, T_m \in N,$$

$$(2) \quad T_x = \text{НОД}(T_u, T_p, T_m).$$

Иначе говоря, T_u / T_x , T_p / T_x , $T_m / T_x \in N$. В этом случае любой момент времени, в котором значения величин в системе существенны для состояния системы, будет иметь вид $t_j = j\tau T_x$, $j = 0, 1, \dots, \frac{T}{\tau T_x}$, а в начале траектории $t = 0$. Зададим переход

системы из состояния, соответствующего времени t_j , в состояние, соответствующее времени t_{j+1} , в виде модели распространения сигнала в полносвязной ИНС прямого распространения с числом входов, на 3 большим числа нейронов (так как f не предсказывается), см. рис. 2.

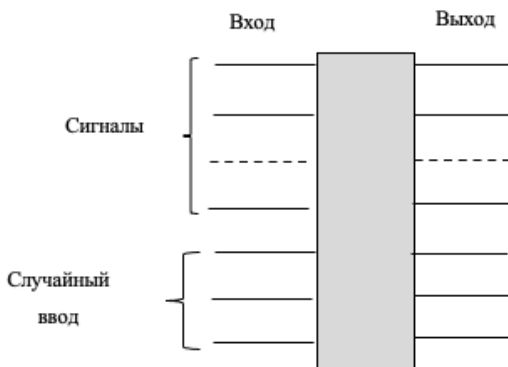


Рис. 2. Первый слой ИНС: переход системы из состояния t_j к состоянию t_{j+1}

Далее будем достраивать слои нейронной сети, сохраняя число сигналов, причем не будем заботиться о вводе внешних воздействий: первый ввод их был случайным, а далее ИНС сама их моделирует для учета в дальнейших оценках позиции и других сигналов, значения которых время от времени становятся известными. Так как сигналы в этой ИНС характеризуют текущее состояние АНПА и управляющие воздействия на него, то можно считать эту систему управления адаптивной: внешние воздействия на нее будут отражаться в сигналах системы позиционирования, а вначале введенное случайное воздействие в конечном счете превратится просто в набор скрытых параметров системы. При должном обучении такая система будет стремиться удерживать АНПА на заданной траектории, если правильно построить целевую функцию.

В начале известны все величины, кроме $f^k(0)$, и движители положим выключенными:

$$(3) \quad u_0^j, p_0^k \equiv x_0^k, m_0^s = 0.$$

Для нахождения весов и смещений ИНС требуются данные для ее обучения. А именно, соответствующие реальности совокупности сигналов u , p , m и меток x такие, что отклонение $|p - x|$ стремится к нулю (уменьшается по сравнению с предыдущим). Иначе говоря, для каких-то значений времени имеются u^k, p^k, m^s

такие, что для более позднего момента времени величина $|p - x|$ меньше, чем для начального.

Ввиду того, что отклонения, вызываемые внешними воздействиями за рассматриваемое время, сравнительно малы, мы положим, что в выходе последнего слоя должно быть $p^k = x^k$ для соответствующего момента времени. Тем самым «усилия» ИНС всегда будут направлены на достижение нужных точек траектории, куда бы вначале ни было направлено судно. И управляющее воздействие будет окончательно определяться именно в последнем слое.

Участок ИНС, в конце которого будут известны все сигналы (кроме трех, изображенных на рис. 1 внизу), определится числом слоев $N = \text{НОК}(T_u, T_p, T_m)$.

В этой N -слойной нейронной сети будут все параметры системы управления в виде ее весов и смещений.

4. Обучение ИНС прямого распространения

Рассмотрим процесс обучения на примере конкретного числа слоев. Пусть $T_u = 2$, $T_p = 4$, $T_m = 3$, $T_x = 1$, $N = 12$. Тогда полносвязная ИНС прямого распространения будет иметь вид, представленный на рис. 3.

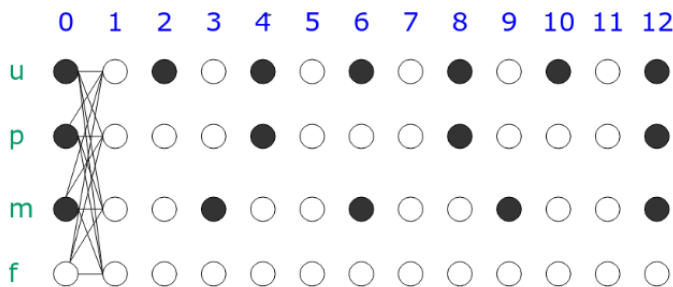


Рис. 3. Частный случай ИНС системы адаптивного управления

Обозначим веса и смещения через W_l^{da} и b_l^n соответственно, где l – номер слоя, n – номер нейрона, d, a – номера нейронов

донора (в предыдущем слое) и акцептора (в текущем слое). Тогда выход любого слоя запишется в виде

$$(4) \quad y_l^a = \delta_l^a \left(\sum_{d=0}^{D-1} W_l^{da} y_{l-1}^d + b_l^a \right),$$

$$(5) \quad l = 1, \dots, N,$$

$$(6) \quad D = J + S + 6,$$

где D – количество признаков, соответствующих разным физическим величинам; J – число моментальных показаний инерциальной навигационной системы; S – количество движителей.

Кодировка каждого слоя будет следующей:

$$(4) \quad y_0^0 = u_0^0, \dots, y_0^{J-1} = u_0^{J-1};$$

$$y_0^J = p_0^0, \dots, y_0^{J+2} = p_0^2;$$

$$y_0^{J+3} = m_0^0, \dots, y_0^{J+2+S} = m_0^{S-1};$$

$$y_0^{J+3+S} = f_0^0, \dots, y_0^{J+5+S} = f_0^2.$$

Инициуруем случайные параметры W_l^{da} и b_l^n . Пройдем ИНС прямо до последнего слоя, получим выходы y_l^a . Размеченные значения выходов будем обозначать через \hat{y} . Начнем обучение с последнего слоя (обратное распространение ошибки).

Коррекция весов последнего слоя ($l = 12$) будет выглядеть так (дельта-правило):

$$(8) \quad \Delta W_l^{da} = -k \sum_a (y_l^a - \hat{y}_l^a) y_l^a (1 - y_l^a) y_{l-1}^d,$$

$$(9) \quad \Delta b_l^a = -k \sum_a (y_l^a - \hat{y}_l^a) y_l^a (1 - y_l^a),$$

где суммирование ведется по всем имеющимся пакетам сигналов.

Каждый пакет сигналов формируется после прохода ИНС. То есть после первого прохода $a = 1$, затем $a = 2$ и т.д. Разница при каждом новом проходе ИНС будет в наборе сигналов на ее входе, а также в наборе меток для обучения в слоях. То есть данные для обучения модель получает прямо из своей работы. Конечно, в начале работы обучение не будет заметно и модель будет вести себя скорее всего неадекватно, но чем дальше будет происходить обучение, тем лучше будет удержание траектории.

Далее увидим это на графиках. В реальности неадекватное поведение не будет заметно, так как длительность этого режима меньше или порядка характерного времени для объекта управления.

Согласно принятому в методе обратного распространения ошибки предположению, разметка слоя $l - 1$ будет иметь вид

$$(10) \hat{y}_{l-1}^d = y_{l-1}^d - \sum_a \delta_l^a W_l^{da},$$

$$(11) \delta_l^a = (y_l^a - \hat{y}_l^a) y_l^a (1 - y_l^a).$$

При обучении ИНС мы будем пользоваться этим правилом только тогда, когда нет реальных данных (в случае прозрачных точек на рис. 2). Если же реальные данные имеются, то будем использовать их вместо соответствующего \hat{y}_{l-1}^d без расчета по обобщенному дельта-правилу. В последнем слое текущего прохода сети в сигналах p всегда будем ставить разметку x . Но при дальнейшем прохождении сети данные для пакета номер $a - 1$ будут содержать реальную позицию p вместо x . Так будет обеспечена подстройка весов сети на большем, чем $N\tau$, промежутке времени. Кроме того, напомним, что для всех пакетов, кроме первого, имеются сигналы и от f (их нет физически, они моделируется нейронной сетью).

В целом сигналы, подаваемые в ИНС, будут иметь следующий вид:

1. Показания инерциальной навигационной системы u^j в количестве J и каждое может принимать ряд значений (их количество равно мощности P_u признака u^j и зависит от кванта и максимального значения).

2. Данные системы позиционирования p^k в виде трех векторов и каждое может принимать ряд значений (их количество равно мощности P_p признака p^j и зависит от кванта и максимального значения).

3. Данные управляющего воздействия m^s в виде допустимых команд на каждый из S движителей.

После этого можно тестировать АНПА: запускать его с разными начальными данными и по ходу его движения он всегда

должен стремиться к желаемой траектории, корректируя направление движителями.

5. Численное моделирование работы системы управления

5.1. ДАННЫЕ И СТРУКТУРЫ ДАННЫХ

Для численного моделирования использовался язык Python 3.10 с библиотеками NumPy (работа с массивами данных) и matplotlib для построения графиков.

Сигналы инерциальной навигационной системы *и* смоделированы в виде шести значений, лежащих в диапазоне от 0 до 9,9 с квантом 0,1, т.е. 6 признаков мощностью 100 каждый. На каждый момент времени, когда эти сигналы доступны, выдаются все 6 значений. Для удобства хранения и работы входящий сигнал сети записывается в один массив с кодировкой в соответствии с формулами (4). Для процесса обучения таких сигналов должно быть много, т.е. мы будем их накапливать при каждом проходе сети. Поэтому этот массив представляет собой матрицу, в каждом векторе данных которой (строке) первые 6 (от 0 до 5) значений – показания инерциальной навигационной системы. Сигналы, получаемые в движении, хранятся в оперативной памяти в виде словарей. Ключом первого словаря является временная метка (номер слоя), значением – второй словарь. Ключом второго словаря является номер сигнала, а значением – массив значений этого сигнала в это время для каждого экземпляра данных.

Сигналы системы позиционирования *p*, когда они доступны, представляют собой три значения. Их диапазон и квант в модели приняты равными соответственно $-100, \dots, 99,99$ и $0,01$. То есть это три признака с мощностями по 20000 каждый. Их записываем аналогично: входящие значения – в массив входящих данных признаками 8, 9, 10. Ясно, что ключи первого словаря не будут охватывать все слои нейронной сети (так как данные появляются не на каждом шаге времени). Для каждого пакета, используемого при обучении, имеется своя такая структура данных.

Сигналы управления двигателями m доступны каждые θ_m единиц времени. В модели для визуализации рассматривается 4 двигателя. В стартовой позиции все двигатели отключены: это четыре сигнала, равных 0. Далее они будут принимать значения от $-0,9$ до $1,0$ с квантом $0,1$. Таким образом, у нас будет 4 сигнала с мощностью 20 каждый. Их хранение осуществляется аналогично вышеуказанным сигналам: входящий – в массиве, текущие – в своих словарях для каждого пакета (т.е. для каждого прохода сети).

Параметризация желаемой траектории задается в виде последовательности точек, т.е. в виде матрицы. Структурой данных для неё является массив. В каждый момент времени (в каждом слое ИНС) кроме момента выхода пакета из ИНС будет присутствовать либо предсказанное значение p , либо значение, пришедшее из навигационной системы, и только на выходе будет использоваться траекторное значение x . Поэтому и значения из словарей на этапе первичного обучения никак не будут взаимодействовать с желаемой траекторией [16]. Грубо говоря, система должна сначала описать в параметрах свое окружение, а уже потом должна обучаться движению по траектории.

5.2. ПРОГРАММНАЯ РЕАЛИЗАЦИЯ

В качестве примера создадим нейронную сеть из 12 слоев по 16 нейронов в каждом. 16 сигналов на входе формируются как 6 сигналов инерциальной навигационной системы, 3 сигнала системы позиционирования, 4 сигнала управляющего воздействия и три сигнала генерируются случайно (только для первого пакета!) – это эмуляция внешних течений f , которые далее полагаются задаваемыми точно такими же параметрами, как u : с теми же диапазонами и квантованием.

В программном коде использовались следующие функции:

- 1) генерации весов и смещений сети;
- 2) прохода слоя вперед;
- 3) проход нескольких слоев вперед;
- 4) коррекции весов и смещений последнего слоя (по меткам выхода нейронной сети);

5) коррекции весов и смещений скрытых слоев (с учетом частично размеченных данных).

Изначально генерируем случайные веса и смещения, затем обучаем нейронную сеть.

Внутри функции обучения основной цикл проходит по каждой эпохе обучения, которая соответствует каждому новому пакету данных. То есть эпоха использует все данные, которые накопились до нее и для нее. Для каждой эпохи проходит цикл по каждому слою ИНС. Обучение каждого слоя методом обратного распространения ошибки с использованием градиентного спуска. Обновленные весовые матрицы и векторы смещений сохраняются.

В результате обучения получаем матрицу весов и смещений, которая содержит информацию о закономерностях движения АНПА.

С помощью библиотеки `matplotlib` производим построение графиков абсолютной величины отклонения аппарата от желаемой траектории в каждый момент времени (рис. 4).

5.3. РЕЗУЛЬТАТЫ И ОБСУЖДЕНИЕ

Моделирование движения аппарата по траектории для испытания работы предложенной системы управления было произведено путем генерации небольших отклонений от заданной траектории, которая, в свою очередь, представляет собой кривую в абсолютной системе координат. В численном моделировании турбулентные эффекты не учитывались, так как предполагалось малое смещение аппарата. Здесь надо сказать, что природа движения объекта для целей управления в данном случае вообще не важна: главное, чтобы во время обучения (менее секунды) и управляющего воздействия (еще меньше) это движение подчинялось каким-то неизменным в этом промежутке времени законам (любым). Воздействие делалось добавками к компонентам скорости, нормально распределенными около 0 со стандартным отклонением 0,1 модуля скорости.

Проведено по три испытания с разными стартовыми сигналами f для систем, обученных на 80 и на 100 эпохах. Как видно на графике на рис. 6, отклонение от траектории в последних 12

шагах после 80 эпох обучения начало уменьшаться с 8-го шага во всех испытаниях.

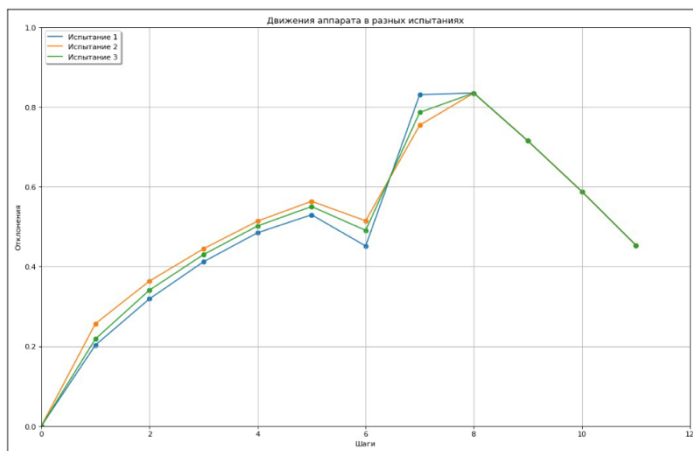


Рис. 4. Абсолютные величины отклонения аппарата от желаемой траектории в каждый момент времени в каждом испытании после 80 эпох обучения

На 3D-графиках рис. 5 и 6 можно увидеть, как смоделированная и обученная соответственно в 80 и 100 эпохах система управления провела аппарат по заданной траектории.

Видно, что на 100 эпохах качество проводки лучше, но в целом и система, обученная в 80 эпохах, установила аппарат на траекторию и удержала даже на значительном искривлении.

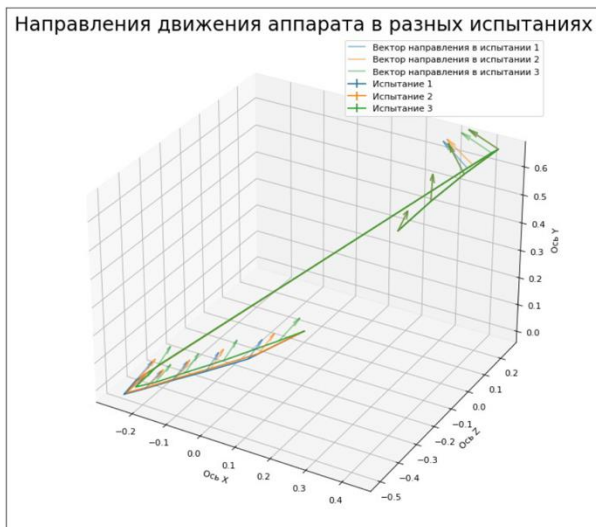


Рис. 5. Траектории движения аппарата в испытаниях с системой, обученной 80 эпохами

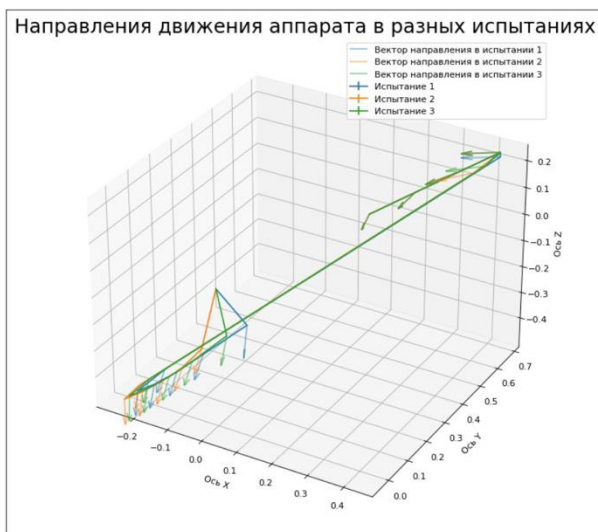


Рис. 6. Траектории движения аппарата в испытаниях с системой, обученной в 100 эпохах

Графики кривых обучения для разных испытаний приведены на рис. 7. Видно, что скорость обучения варьируется в зависимости от стартовых условий, но незначительно. На графике представлены только нейроны, отвечающие за данные инерциальной навигационной системы, внешние и управляющие воздействия, так как позиционирование уже проиллюстрировано выше.

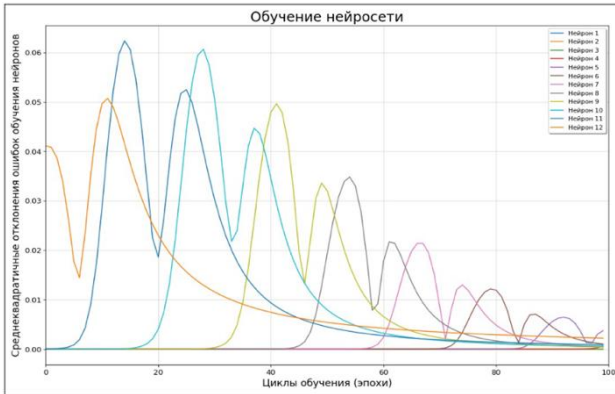


Рис. 7. Процесс обучения ИНС

6. Заключение

В настоящем исследовании была предложена и смоделирована интеллектуальная система адаптивного управления для АНПА с заданной целью – удержание аппарата на заданной траектории.

Ключевой идеей работы было использование простой ИНС, которая эксплуатируется как онлайн-обучаемая система: она выдает требуемую информацию об управляющих воздействиях и обучается одновременно. Для обучения используются накапливающиеся данные о работе АНПА и системы управления.

В результате численного моделирования работы системы управления предложенного типа были получены срезы моделированных и целевых траекторий движения АНПА, представленные на графиках.

Полученный результат показывает, что даже сравнительно простые системы искусственного интеллекта при определенной

стратегии обучения способны достигать хороших результатов в задачах управления. Так, в модели 12-слойной ИНС обучение достигло устойчивых результатов через 100 эпох. То есть понадобилось прохождение 1200 слоев. Поскольку каждый слой эквивалентен промежутку времени минимальной дискретности, который может соответствовать сотым долям секунды, обучение практически было закончено примерно за секунду. Но конкретные значения временных параметров зависят от объекта управления и условий его движения.

Литература

1. БОРЕЙКО А.А., ИНЗАРЦЕВ А.В., МАШОШИН А.И., ПАВИН А.М. и др. *Система управления АНПА большой автономности на базе мультиагентного подхода* // Подводные исследования и робототехника. – 2019. – Т. 28, №2 – С. 23–31.
2. РАШИД Т. *Создаем нейронную сеть*. – СПб. : ООО «Диалектика», 2019. – 272 с.
3. ЦЫПКИН Я.З. *Адаптация и обучение в автоматических системах*. – М.: Гл. ред. физ.-мат. лит-ры изд-ва «Наука», 1968. – 400 с.
4. ABOUSSALAH A.M., LEE C. *Continuous control with Stacked Deep Dynamic Recurrent Reinforcement Learning for portfolio optimization* // Expert Systems with Applications. – 2020. – Vol. 140. – P. 1–11.
5. CARLUCHO I., PAULA M.D., ACOSTA G.G. *Double Q-PID algorithm for mobile robot control* // Expert Systems with Applications. – 2019. – Vol. 137. – P. 292–307.
6. SHE G., YU Z. *Neural-network estimators based fault-tolerant tracking control for AUV via ADP with rudders faults and ocean current disturbance* // Neurocomputing. – 2020. – Vol. 411. – P. 442–454.
7. *Deep Neural Evolution* / Eds.: H. Iba, N. Noman // Springer Singapore. – 2020. – 438 p. – ISBN 978-981-15-3685-4 (electronic).
8. DEPTULA P., BELL Z.I., DOUCETTE E.A., CURTIS J.W. et al. *Data-based reinforcement learning approximate optimal con-*

- trol for an uncertain nonlinear system with control effectiveness faults // Automatica. – 2020. – Vol. 116. – P. 1–10.*
9. ELHAKI O., SHOJAEI K. *A robust neural network approximation-based prescribed performance output-feedback controller for autonomous underwater vehicles with actuators saturation // Engineering Applications of Artificial Intelligence – 2020. – Vol. 88. – P. 1–16.*
 10. ELHAKI O., SHOJAEI K. *Neural network-based target tracking control of underactuated autonomous underwater vehicles with a prescribed performance // Ocean Engineering. – 2018. – Vol. 167. – P. 239–256.*
 11. GLATT R., SILVA F.L., BIANCHI R.A., COSTA A.H. *DECAF: Deep Case-based Policy Inference for knowledge transfer in Reinforcement Learning // Expert Systems with Applications. – 2020. – Vol. 156. – P. 1–13.*
 12. HAN H., WEI Y., YE X., LIU W. *Modeling and fuzzy decoupling control of an underwater vehicle-manipulator system // IEEE Access. – 2020. – Vol. 8. – P. 18962–18983.*
 13. HUANG Z, BAI W., LI T., LONG Y. et al. *Adaptive reinforcement learning optimal tracking control for strict-feedback nonlinear systems with prescribed performance // Information Sciences. – 2023. – Vol. 621. – P. 407–423.*
 14. JAIN A., THOKE A.S., PATEL R.N. *Fault classification of double circuit transmission line using artificial neural network // Int. Journal of Electrical Systems Science and Engineering. – 2008. – Vol. 1. – P. 230–235.*
 15. LEI M. *Nonlinear diving stability and control for an AUV via singular perturbation // Ocean Engineering. – 2020. – Vol. 197. – P. 1–11.*
 16. LEWIS F.L., VRABIE D.L. *Reinforcement learning and adaptive dynamic programming for feedback control // IEEE Circuits and Systems magazine. – 2009. – Vol. 9. – P. 32–50.*
 17. MAHANTY R.N., DUTTA GUPTA P.B. *Comparison of Fault Classification Methods Based on Wavelet Analysis and ANN // Electric Power Components and Systems. – 2006. – Vol. 34 – P. 47–60.*
 18. MONTGOMERY D.C. *Design and Analysis of Experiments. – John Wiley & Sons, Inc, 1997. – P. 757.*

19. NEVES G., RUIZ M., FONTINELE J., OLIVEIRA L. *Rotated object detection with forward-looking sonar in underwater applications* // Expert Systems with Applications. – 2020. – Vol. 140. – P. 1–18.
20. PENG Z., ZHAO Y., HU J., GHOSH B.K. *Data-driven optimal tracking control of discrete-time multi-agent systems with two-stage policy iteration algorithm* // Information Sciences. – 2020. – Vol. 481. – P. 189–202.
21. RASHIDI A.J., KARIMI B., KHODAPARAST A. *A constrained predictive controller for AUV and computational optimization using Laguerre functions in unknown environments* // Int. J. Control Autom. Syst. – 2020. – Vol. 18. – P. 753–767.
22. RODRÍGUEZ J., CASTAÑEDA H., GORDILLO J.L. *Lagrange modeling and navigation based on quaternion for controlling a micro AUV under perturbations* // Robotics and Autonomous Systems. – 2020. – Vol. 124. – P. 1–12.
23. SHEN C., SHI Y. *Distributed implementation of nonlinear model predictive control for AUV trajectory tracking* // Automatica. – 2020. – Vol. 115. – P. 1–9.
24. SU H., ZHANG H., SUN S., CAI Y. *Integral reinforcement learning-based online adaptive event-triggered control for non-zero-sum games of partially unknown nonlinear systems* // Neurocomputing. – 2020. – Vol. 377. – P. 243–255.
25. SUN Y., XIANGRUI R., ZHANG G., XIANGBIN W., XU H. *AUV path following controlled by modified Deep Deterministic Policy Gradient* // Ocean Engineering. – 2020. – Vol. 210. – P. 1–14.
26. SURYENDU C., SUBUDHI B. *Formation control of multiple autonomous underwater vehicles under communication delays* // IEEE Trans. on circuits and systems II: Express briefs. – 2020. – Vol. 67, No. 12. – P. 3182–3186.
27. TENG M., YE L., YUXIN Z., YANQING J. et al. *An AUV localization and path planning algorithm for terrain-aided navigation* // ISA Transactions. – 2020. – Vol. 103. – P. 215–227.
28. VAMVOUDAKIS K.G., LEWIS F.L. *Online actor-critic algorithm to solve the continuous-time infinite horizon optimal control problem* // Automatica. – 2010. – Vol. 46. – P. 878–888.

29. WANG X., ZERR B., THOMAS H., CLEMENT B. et al. *Pattern formation of multi-AUV systems with the optical sensor based on displacement-based formation control* // Int. Journal of Systems Science. – 2020. – Vol.51:2. – P. 348–367.
30. WEN G., CHEN C.L., LI W.N. *Simplified optimized control using reinforcement learning algorithm for a class of stochastic nonlinear systems* // Information Sciences. – 2020. – Vol. 517. – P. 230–243.
31. *What's New in Artificial Intelligence from the 2022 Gartner Hype Cycle* [Электронный ресурс] // Gartner. – URL: <https://tinyurl.com/4z4vrhcy> (дата обращения: 26.06.2023).
32. XIA Y., XU K., WANG W., XU G. et al. *Optimal robust trajectory tracking control of a X-rudder AUV with velocity sensor failures and uncertainties* // Ocean Engineering. – 2020. – Vol. 198. – P. 1–16.
33. YAN X-P., WANG S-W., MA F., LIU Y-C. et al. *A novel path planning approach for smart cargo ships based on anisotropic fast marching* // Expert Systems with Applications. – 2020. – Vol. 159. – P. 1–16.
34. YUAN H, SHUAI C., MA J., FANG Y. *An efficient control allocation algorithm for over-actuated AUVs trajectory tracking with fault-tolerant control* // Ocean Engineering. – 2023. – Vol. 273. – P. 113976.
35. ZHENG Z., RUAN L., ZHU M., GUO X. *Reinforcement learning control for underactuated surface vessel with output error constraints and uncertainties* // Neurocomputing. – 2020. – Vol. 399. – P. 479–490.

ADAPTIVE TRAJECTORY CONTROL SYSTEM AUV BASED ON A DIRECT PROPAGATION NEURAL NETWORK

Valeria Romanova, V.I. Vernadsky Crimean Federal University, Simferopol, student (lero4ka2004ro@gmail.com).

Sergei Zuev, Belgorod State Technological University named after V.G. Shukhov, Belgorod, Candidate of Physical and Mathematical Sciences, Associate Professor (sergey.zuev@bk.ru).

Abstract. The paper is devoted to the development and study of an intelligent system of adaptive automatic control with a given target based on the use of artificial neural network of forward propagation. The control object is an autonomous unmanned underwater vehicle (AUV). In this paper, it is proposed to feed the signals received from the systems of the AUV to the input of the neural network, and use the output signal of the neural network for control to keep the vehicle on a given trajectory. As a result of this work, a model and a learning method are proposed that lead to holding the ANPA on a given trajectory under an external influence with a natural constraint for the considered mobile robot. Given a continuous preset trajectory and discrete signals from the ANPA systems, this allows following the preset trajectory with a simple intelligent control system that does not require large computational power. The proposed method of control system training allows pre-training on a numerical model of vehicle motion with random external influences, but does not require such pre-training under certain conditions. It is shown, in particular, that in the case of a sufficiently large learning rate, the model has time to rearrange itself and reacts to changed circumstances. The proposed intelligent system of adaptive automatic control can find application in those cases when the characteristic time of changes in the system is of the order of the training time, and the trajectory of motion satisfies the requirements stated in the paper.

Keywords: artificial neural network, autonomous unmanned underwater vehicle, adaptive control system, intelligent control system, artificial neural network training.

УДК 004.8

ББК 32.813.5

DOI: 10.25728/ubs.2024.108.11

*Статья представлена к публикации
членом редакционной коллегии Б.Р. Андриевским.*

*Поступила в редакцию 21.10.2023.
Опубликована 31.03.2024.*