

На правах рукописи



**КАШКО Василий Васильевич**

**АДАПТИВНАЯ СИСТЕМА УПРАВЛЕНИЯ  
ИНТЕЛЛЕКТУАЛЬНЫМ АГЕНТОМ НА ОСНОВЕ  
ГЛУБОКОГО ОБУЧЕНИЯ С ПОДКРЕПЛЕНИЕМ**

Специальность: 2.3.1. Системный анализ, управление и обработка информации,  
статистика

**АВТОРЕФЕРАТ**

диссертации на соискание учёной степени  
кандидата технических наук

Воронеж – 2026

Работа выполнена в федеральном государственном бюджетном образовательном учреждении высшего образования «Воронежский государственный технический университет»

Научный руководитель: **Олейникова Светлана Александровна**  
доктор технических наук, профессор

Официальные оппоненты: **Варламов Олег Олегович**, доктор технических наук, профессор, ФГАОУ ВО «Московский государственный технический университет им. Н.Э. Баумана», профессор кафедры ИУ-5 «Системы обработки информации и управления»

**Преображенский Андрей Петрович**, доктор технических наук, профессор, ВИВТ-АНОО ВО, руководитель научного клуба

Ведущая организация: **ФГБОУ ВО «Кубанский государственный технологический университет»**

Защита состоится «15» сентября 2026 года в 14<sup>00</sup> часов в конференц-зале на заседании объединенного диссертационного совета 99.2.031.03, созданного на базе ВГТУ, ВГУ и ЛГТУ, по адресу: 394026, г. Воронеж, Московский проспект, 14, аудитория 216.

С диссертацией можно ознакомиться в научно-технической библиотеке ФГБОУ ВО «Воронежский государственный технический университет» и на сайте <https://cchgeu.ru/>

Автореферат разослан «10» июля 2026 г.

Учёный секретарь  
диссертационного совета 99.2.031.03,  
доктор технических наук, профессор



Белецкая  
Светлана Юрьевна

## ОБЩАЯ ХАРАКТЕРИСТИКА РАБОТЫ

**Актуальность темы.** Одной из центральных проблем в современной теории управления, в особенности при работе с динамически изменяющимися неструктурированными средами в контексте киберфизических систем, является задача построения системы управления интеллектуальным агентом. Классические подходы, базирующиеся на точных моделях окружающих сред и предварительно определённых алгоритмах, зачастую оказываются недостаточно эффективными для управления агентом в сложных, динамически изменяющихся или заранее не полностью определённых средах эксплуатации. Применение глубокого обучения с подкреплением представляет собой альтернативу, обеспечивающую киберфизическую систему возможностью самостоятельно обучаться и реализовывать адаптацию к изменениям в среде посредством опыта, получаемого от взаимодействия с окружающей средой на основе метода проб и ошибок.

Проблематика управления интеллектуальным агентом входящем в состав киберфизической системы активно развивается с 1960-х годов. В настоящее время значимый вклад в развитие направления внесло применение глубокого обучения с подкреплением, что в значительной степени повысило автономность и адаптивность к динамически изменяющимся условиям. В частности, задаче обучения множеству стратегий (Multitask/Multi-skill) посвящены труды X. В. Peng, P. Abbeel, S. Levine и др. Обеспечением устойчивости и адаптации (Robust control) занимались J. Hwangbo, M. Hutter, G. Marcus, L. Righetti и др. В работах J. Peters, M. Hutter, C. Finn, S. Laura, S. Levine рассматривается возможность создания универсальных стратегий и переноса обученного в симуляции агента на реальную систему. Среди отечественных учёных, внёсших значимый вклад в изучение управления киберфизическими системами, в частности на основе алгоритмов обучения с подкреплением, являются А.И. Панов, Р.В. Мещеряков, А.П. Преображенский и др.

Однако не все проблемы в данной области решены. В частности, среди проблем обучения выделяются забывание при добавлении новых задач, конфликт градиентов между навыками, ухудшение качества при масштабировании и сложность балансировки между общими и специфичными представлениями. Кроме того, при переносе агента из симуляции на реальные киберфизические системы возникает сложность с обеспечением одновременного решения базовых задач, нестабильность обучения из-за роста, отсутствие масштабируемых архитектур для универсальных политик, высокие трудозатраты валидации на реальных системах.

В результате, существующие подходы либо демонстрируют ограниченную обобщающую способность, либо страдают от деградации при переносе обученного агента из симуляции. Таким образом, проведённое

исследование отвечает современным вызовам в области построения адаптивных самообучающихся систем управления и вносит вклад в решение актуальной научной проблемы построения системы управления интеллектуальным агентом на базе глубокого обучения с подкреплением.

**Целью работы** является разработка адаптивной системы управления интеллектуальным агентом на основе обучения с подкреплением, обеспечивающей обучение множеству стратегий с минимизацией коллизий на опыте, полученном при непосредственном взаимодействии киберфизической системы с окружающей средой и автономную адаптацию под изменения среды на примере решения задачи управления локомоцией шагающего робота.

**Задачи исследования.** Достижение поставленной цели диссертационного исследования требует решения следующих задач:

1. Произвести анализ существующих подходов в области управления интеллектуальным агентом на основе глубокого обучения с подкреплением, включая проблематику управления локомоцией шагающих роботов как частного случая киберфизических систем с целью определения наилучшего алгоритма обучения с подкреплением.

2. Разработать математическое обеспечение системы управления интеллектуальным агентом, обеспечивающее повышение уровня безопасности и устойчивости функционирования управляемой киберфизической системы.

3. Разработать специальное алгоритмическое обеспечение системы управления интеллектуальным агентом, обеспечивающее ускорение обучения и переноса агента на реальную систему за счёт увеличения времени устойчивого функционирования и сокращения числа нештатных режимов (коллизий).

4. Разработать архитектуру глубокой нейронной сети, обеспечивающей агента, основанного на глубоком обучении с подкреплением, возможностью обучения нескольким стратегиям в контексте одной глубокой нейронной сети и ликвидирующей эффект «забывания» и потерю деталей каждой из стратегий.

5. Разработать структуру системы управления интеллектуальным агентом, обеспечивающую автономное адаптивное безопасное обучение множеству стратегий с последующим переносом на реальную киберфизическую систему и безопасной настройкой на примере формирования локомоции шагающего робота.

6. Произвести экспериментальное исследование эффективности разработанного математического и алгоритмического обеспечения, архитектуры многофункционального агента и структуры системы управления.

**Объект исследования:** система управления интеллектуальным агентом, основанная на глубоком обучении с подкреплением, входящая в состав киберфизической системы.

**Предмет исследования:** математическое и алгоритмическое

обеспечение системы управления интеллектуальным агентом на примере генерации шагающей локомоции киберфизической системы шагающего робота, включая формализацию задачи управления, методы обучения с подкреплением, архитектурные решения по построению нейросетевого агента и алгоритмы формирования и использования множества стратегий в условиях непосредственного взаимодействия с окружающей средой.

**Методы исследования.** В процессе решения задач диссертационного исследования использовались методы: системного анализа, принятия решений, теории обучения с подкреплением, математического моделирования, теории автоматического управления, структурного анализа, теории глубокого обучения, теории Марковских процессов принятия решений, динамического программирования, теории вероятностей и математической статистики, методов оптимизации, имитационного моделирования, объектно-ориентированного программирования и проектирования.

**Тема диссертационного исследования** соответствует следующим пунктам паспорта специальности 2.3.1 «Системный анализ, управление и обработка информации, статистика»: п.5 «Разработка специального математического и алгоритмического обеспечения систем анализа, оптимизации, управления, принятия решений, обработки информации и искусственного интеллекта»; п.9 «Разработка проблемно-ориентированных систем управления, принятия решений и оптимизации технических объектов»; п.10 «Методы и алгоритмы интеллектуальной поддержки при принятии управленческих решений в технических системах».

**Научная новизна работы.** В контексте диссертационного исследования были получены следующие результаты, характеризующиеся научной новизной:

1. Математическое обеспечение системы управления интеллектуальным агентом, отличающееся внедрением в агента возможности предсказания риска возникновения нестабильных состояний, основанной на выделенном множестве критических состояний, и обеспечивающее повышение безопасности обучения и стабилизацию функционирования управляемой киберфизической системы.

2. Специальное алгоритмическое обеспечение системы управления интеллектуальным агентом, отличающееся фокусированием обучения на границе устойчивости, обеспечивающее повышение безопасности обучения и переноса агента с последующей безопасной настройкой в реальной среде.

3. Архитектура глубокой нейронной сети агента, отличающаяся возможностью аппроксимации множества стратегий в рамках одной полносвязной нейронной сети, и обеспечивающая реализацию многофункционального управления с ликвидацией эффекта «забывания» и потери деталей каждой из стратегий.

4. Структура системы управления, базирующаяся на нейросетевом кластере с переключающимися нейронами, обеспечивающая автономное

адаптивное обучение множеству стратегий с последующей безопасной настройкой в реальной среде.

**Практическая значимость** исследования заключается в создании и экспериментальной апробации проблемно-ориентированной системы управления интеллектуальным агентом, отвечающим за формирование шагающей локомоции киберфизической системы шагающего робота, реализующей разработанные алгоритмы и архитектурные решения. Разработанная система обеспечивает адаптивное управление движением в условиях неопределенности внешней среды. Предложенные решения целесообразно использовать при разработке и управлении робототехническими системами, предназначенными для функционирования в изменяющихся условиях, а также в образовательных и исследовательских комплексах.

#### **Положения, выносимые на защиту**

1. Математическое обеспечение системы управления интеллектуальным агентом обеспечивает повышение безопасности обучения и стабилизацию функционирования киберфизической системы за счет внедрения в агента возможности предсказания риска возникновения нестабильных состояний, основанной на выделенном множестве критических состояний.

2. Алгоритмическое обеспечение системы управления интеллектуальным агентом позволяет повысить безопасность обучения и переноса агента с последующей безопасной настройкой в реальной среде.

3. Архитектура глубокой нейронной сети агента решает проблему низкой репрезентативной способности сети и обеспечивает реализацию многофункционального управления с ликвидацией эффекта «забывания» и потери деталей каждой из стратегий за счет их аппроксимации в рамках одной полносвязной нейронной сети.

4. Структура системы управления обеспечивает автономное адаптивное безопасное обучение множеству стратегий с последующей безопасной настройкой в реальной среде за счет использования нейросетевого кластера с переключающимися нейронами и учёта критических состояний.

**Результаты внедрения.** Разработанные в диссертационном исследовании схемы и механизмы были внедрены в образовательный процесс ФГБОУ ВО «Воронежский государственный технический университет» в виде лабораторного практикума и лекционного курса по дисциплинам "Интеллектуальные системы", "Системы искусственного интеллекта". Кроме того, алгоритмическое и программное обеспечение для построения адаптивных систем управления интеллектуальными агентами, основанного на методах обучения с подкреплением внедрены в деятельность компании ООО «Девелоперс». Эффект от внедрения заключается в повышении точности принятия решений в изменяющихся условиях и сокращении времени на ручную настройку параметров алгоритмов

управления за счёт автоматического обучения.

**Апробация работы.** Рассмотрение и обсуждение основных положений диссертационного исследования проводились в контексте следующих научно-практических конференций: VI Всероссийской научно-практической конференции «Современные информационные технологии. Теория и практика» (Череповец, 2023), XV Международная интернет-конференция молодых учёных, аспирантов и студентов «Инновационные технологии: теория, инструменты, практика» (Пермь, 2023), Международного форума профессионального образования «Антропоцентрические науки в образовании: вызовы, трансформации, ресурсы» (Воронеж, 2024), Международной научной конференции «Актуальные проблемы прикладной математики, информатики и механики» (Воронеж, 2024), Международной научно-практической конференции «Управление программным инжинирингом» (Воронеж, 2025, 2026), Международной научно-практической конференции «Интеллектуальные информационные системы» (Воронеж, 2025), Международной научной конференции «Актуальные проблемы прикладной математики, информатики и механики» (Воронеж, 2025, 2026), XVII Международная интернет-конференция молодых учёных, аспирантов и студентов «Инновационные технологии: теория, инструменты, практика» (Пермь, 2025), II Научно-практическая конференция «Интеллектуальные технологии цифровой инженерии» (Воронеж, 2026).

**Публикации.** В результате выполнения диссертационного исследования было опубликовано 19 научных работ, включая 6 – в изданиях ВАК РФ и 1 свидетельство о регистрации программы. В опубликованных в соавторстве работах, перечень которых приведён в конце автореферата, автором персонально был получен следующий перечень результатов: [9, 10, 11] – теоретический анализ существующих подходов и алгоритмов обучения с подкреплением, применяемых для реализации управления движением шагающих роботов, [1, 12, 16] – математическое обеспечение системы управления локомоцией шагающего робота, отличающееся применением выделенного множества критических состояний и обеспечивающее безопасное обучение, и последующий безопасный перенос обученного агента на реальную систему; [2, 3, 5, 13,14,15, 20] – экспериментальное исследование специфики существующих и эффективности применения разработанных алгоритмов управления локомоцией шагающего робота; [4, 17, 18] – алгоритмическое обеспечение управления локомоцией шагающего робота; [6] – построение архитектуры нейронной сети, обеспечивающей агента, основанного на глубоком обучении с подкреплением, многофункциональностью в контексте одной полносвязной нейронной сети; [19] – построение многофункционального агента на базе полносвязной сети-кластера с использованием переключающих нейронов.

**Структура и объем работы.** В состав диссертационной работы входят: введение, четыре главы, заключение, список литературы, состоящий из 167 наименований. Изложение основной части представлено на 167 страницах, с

использованием 20 рисунков и 4 таблиц.

## ОСНОВНОЕ СОДЕРЖАНИЕ РАБОТЫ

**Во введении** произведено обоснование актуальности производимого исследования, осуществлена постановка цели, идентифицированы изучаемый объект и предмет. Сформулированы пункты научной новизны и обозначена практическая значимость результатов. Приведены сведения о внедрении и апробации работы.

**В первой главе** производится анализ проблемы построения системы управления интеллектуальным агентом на основе обучения с подкреплением для формирования шагающей локомоции киберфизической системы шагающего робота, и методов её решения. В результате был реализован следующий дизайн исследования, представленный на рис. 1.

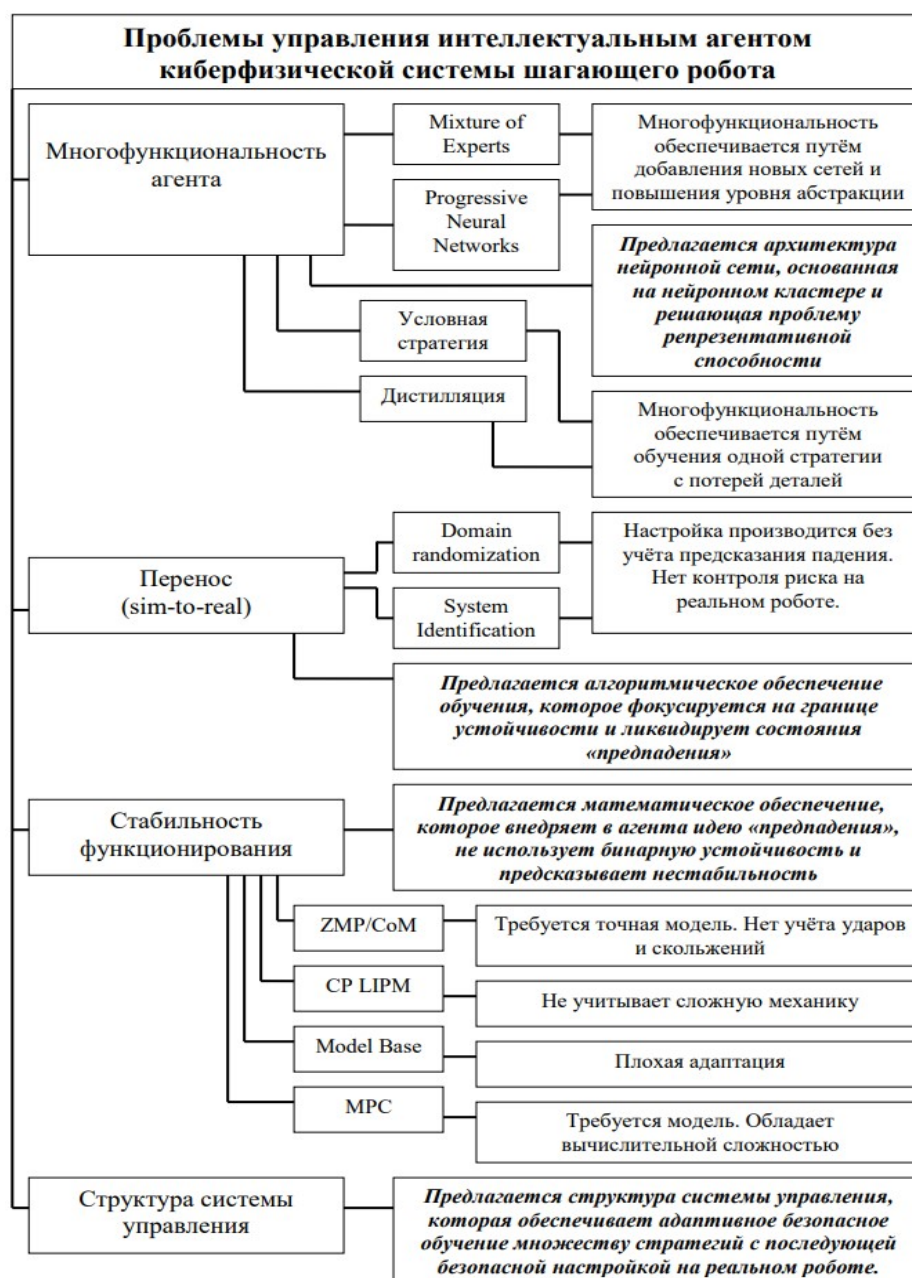


Рис. 1 – Дизайн исследования

В заключение главы приводится формулировка целей и задач исследования.

**Вторая глава** посвящена построению математического обеспечения системы управления интеллектуальным агентом, обеспечивающего безопасное обучение за счёт минимизации риска возникновения неустойчивого состояния. Производится постановка задачи формирования шагающей локомоции, производимой киберфизической системой шагающего робота, используемой в качестве тестового стенда. Её формулировка звучит следующим образом. Пусть некоторая робототехническая шагающая платформа, обладающая двумя или более конечностями (педипуляторами) шарнирного типа, помещена в заранее не детерминированную среду, которую в определённый момент времени можно охарактеризовать кривизной и шероховатостью рельефа, вязкостью грунта. Перед устройством ставится задача самообучения выполнению указанной локомоторной программы из множества допустимых на опыте непосредственного взаимодействия со средой и адаптации полученной стратегии под разнообразные типы поверхностей. Вводятся допущения. В контексте исследования рассматривается одна локомоторная программа на разных типах рельефа местности. Локомоция задаётся вручную. Минимальный необходимый набор сенсоров состоит из гироскопа-акселерометра, датчиков удара и касания поверхности земли педипуляторами. Построение локомоции рассматривается при низких скоростях перемещения.

Пусть общее число сочленений робота равно  $N$ . Тогда состояние механизма в момент времени  $t$  представимо в виде:

$$d_t = (\theta_1, \theta_2, \theta_3, \dots, \theta_N), \quad (1)$$

где  $N$  – количество сочленений робота,  $\theta_i$  – угол (в радианах)  $i$ -ого сочленения,  $d_t$  – состояние робота в момент времени  $t$ .

Следовательно, реализация управления перемещением механизма осуществляется через поочерёдное чередование кортежей в каждый дискретный момент времени  $t$ . Другими словами, вектор  $d$  – выходная шина управляющего устройства.

На вход системы управления передаётся предыдущее состояние механизма  $d_{t-1}$  и угловые скорости приводов  $\dot{d}_t$ , обеспечивающие демпфирование, в процессе локомоции. Для ликвидации шумов, скорости подвергаются фильтрации. Определение ориентации робота в пространстве и удержание баланса обеспечивается за счёт показаний датчика гироскопа-акселерометра, на выходах которого получают измеренные углы тангажа –  $\eta$ , крена –  $\mu$ , рысканья –  $\sigma$  и ускорения  $a_x, a_y, a_z$ . Для фиксации пространственного перемещения механизма требуются координаты точки центра тяжести  $v_x^{CoM}$  и  $v_y^{CoM}$ . При движении необходимо учитывать конечности, расположенные на земле. Для этого используются датчики

касания конечностей, в совокупности генерирующие вектор касания  $T = (T_1, T_2, \dots, T_N)$ . Для обозначения факта завершения эпизода применяется бинарный флаг  $f$ , принимающий значения из множества  $F: \{0,1\}$ . В результате интерфейс системы управления интеллектуальным агентом, формирующей шагающую локомоцию робота имеет следующий вид:

$$IN: (d_t, d_{t-1}, \dot{d}_t, v_{CoM_x}^t, v_{CoM_y}^t, a_x, a_y, a_z, \eta, \mu, \sigma, T_t, f) \quad OUT: (d_{t+1}) \quad (2)$$

Рассматривается динамика системы управления. Пусть  $S \subseteq R^J$  – пространство реальных состояний системы, где  $s_t \in S$  – состояние среды в момент времени  $t$ . Пространство наблюдений  $X \subseteq R^I$ , где  $x_t \in X$  – наблюдение среды в момент времени  $t$ , является подпространством множества реальных состояний  $S$ . Задано множество всех доступных агенту действий  $A \subseteq R^K$ , где действие агента в момент времени  $t$  –  $a_t \in A$ . Величины  $I, J, K \in N$  – мощности множеств  $R^I, R^J, R^K$  и  $t = 1, \dots, \infty$ . В результате, функция перехода из текущего состояния  $s_t$  в новое состояние  $s_{t+1}$ , под влиянием выбранного действия  $a_t$ , с реализацией последующего преобразования его в наблюдение  $x_t$ , имеет следующий вид:

$$\begin{aligned} s_{t+1} &= g(s_t, a_t) \\ x_t &= h(s_t) \end{aligned} \quad (3)$$

где  $g: R^J \times R^K \rightarrow R^J$  – функция перехода в контексте реальных состояний системы,  $h: R^J \rightarrow R^I$  – функция отображения реального состояния на наблюдение.

Наблюдение разрабатываемой системы управления, в контексте задачи управления локомоцией, имеет следующий вид:

$$x_t = (d_t, d_{t-1}, \dot{d}_t, v_{CoM_x}^t, v_{CoM_y}^t, a_x, a_y, a_z, \eta, \mu, \sigma, T_t, f), \quad (4)$$

Рассматривается формализация управления интеллектуальным агентом в рамках Марковского процесса. Исходя из постановки тестовой задачи, Марковский процесс принятия решений имеет следующий вид:

$$\begin{aligned} s_t &= (d_t, \dot{d}_t, \omega_t, v_t, c_t) \\ a_t &= \Delta d_t \\ s_{t+1} &\sim P(s_t, a_t) \\ R_t &= R(s_t, a_t) \\ \pi &= \arg \max E[\sum \gamma^t R_t] \end{aligned} \quad (5)$$

где  $s_t$  – текущее состояние среды,  $d_t$  – углы сочленений (текущее состояние робота),  $\dot{d}_t$  – угловые скорости сочленений,  $\omega_t$  – угловая скорость робота,  $v_t$  – линейное ускорение,  $c_t$  – контакты конечностей с поверхностью земли,  $a_t$  – действие,  $\pi$  – стратегия.

В контексте исследования предлагается ввести множество критических состояний. *Критическое состояние* представляет собой состояние, при котором киберфизическая система находится на грани достижения неустойчивого состояния. В контексте частного случая решения задачи локомоции шагающего робота *критическое состояние* представляет собой состояние, при котором робот ещё устойчив, но находится на грани потери баланса или близок к механическим повреждениям узлов. Множество критических состояний образует подмножество пограничных состояний  $S_{critical}$ , расположенных на границе, между гарантированно устойчивыми состояниями и неустойчивыми.

$$S = S_{safe} \cup S_{critical} \cup S_{fail} \quad (6)$$

где  $S_{safe}$  – множество устойчивых состояний,  $S_{fail}$  – неустойчивые состояния, приводящие к падению или повреждению,  $S_{critical}$  – пограничные состояния, близкие к неустойчивым.

Оно может рассматриваться как подпространство миварной модели знаний, используемое для формирования ограничений при выборе управляющих воздействий.

Введение множества критических состояний реализуется в виде аппроксимации границы устойчивости системы, применяемой для ограничения поведения стратегии. В результате осуществляется переход от набора порогов к формальному множеству критических состояний. Производится явное моделирование границы устойчивости, используемое в обучении агента. Для измерения расстояния до множества неустойчивых состояний вводится скалярная функция риска, которая принимает значения из диапазона  $[0,1]$  и характеризует меру близости к потере устойчивости:

$$C : S \rightarrow R_{\geq 0}, \quad (7)$$

где  $S$  – множество состояний,  $R_{\geq 0}$  – некоторое множество неотрицательных действительных чисел.

Чем меньше значение функции  $C(s)$ , тем безопаснее состояние. Множество критических состояний определяется, как:

$$\begin{aligned} S_{critical} &= \{s \in S_{safe} \mid C(s) \geq C_{crit}\} \\ S_{critical} &= \partial S_{safe} \end{aligned} \quad (8)$$

где  $C(s)$  – функция близости к неустойчивости,  $C_{crit}$  – порог критичности, показывающий расстояние до неустойчивых состояний, относительно которого состояние считается принадлежащим множеству критических.

Функция близости к неустойчивости может быть задана, как эвристически (на основе показаний сенсоров), так и быть аппроксимирована посредством нейронной сети. Эвристическое определение функции  $C(s)$  имеет следующий вид:

$$C(s) = \sum_{i=0}^N \omega_i C_i(s), \quad (9)$$

где  $C_i(s)$  – источник риска с индексом  $i$ ,  $\omega_i$  – весовой коэффициент важности  $i$ -ого риска,  $N$  – количество рисков.

Под источником риска понимается некоторая скалярная функция  $C$ , коррелирующая с риском возникновения неустойчивого состояния. В контексте задачи формирования локомоции соответствует риску потери устойчивости или повреждения.

С учётом концепции критических состояний и функции близости к неустойчивости, задача обучения с подкреплением примет следующий вид:

$$\begin{aligned} E_{\pi_\theta} \left[ \sum_{t=0}^T \gamma^t r(s_t, a_t) \right] &\rightarrow \max, \\ E_{\pi_\theta} [C(s_t)] &\leq \delta \end{aligned} \quad (10)$$

где  $\delta$  – порог риска (задаётся экспериментально).

Преобразовав задачу (10) путём создания Лагранжиана, получим:

$$L(\theta, \lambda) = E_{\pi_\theta} \left[ \sum_{t=0}^T \gamma^t r(s_t, a_t) \right] - \lambda (E_{\pi_\theta} [C(s_t)] - \delta) \quad (11)$$

где  $\lambda$  – множитель Лагранжа.

Вводится дополнительный штраф в вознаграждение:

$$r'_t = r_t - \lambda C(s_t) \quad (12)$$

Завершение эпизода производится в соответствии со следующей функцией:

$$end = \begin{cases} s_t \in S_{fail} \\ C(s_t) > C_{max} \end{cases}, \quad (13)$$

где  $C_{\max}$  – жёсткий порог критичности.

В результате, на основании функции  $C(s)$  можно концептуально представить состояния следующим образом:

$$\begin{aligned} S_{safe} &: C(s) < C_{crit} \\ S_{critical} &: C_{crit} \leq C(s) < C_{\max} \\ S_{fail} &: C(s) \geq C_{\max} \end{aligned} \quad (14)$$

Для тестирования была выбрана шагающая платформа с двумя педипуляторами. Решение прямой задачи кинематики робота производится на основе группы преобразований SE(3) (Special Euclidean Grope in 3D) – специальной евклидовой группы в трёх измерениях, представляющей множество всех возможных жёстких преобразований в трёхмерном пространстве и имеющей следующий вид:

$$T = \begin{bmatrix} R & p \\ 0 & 1 \end{bmatrix}, \quad (15)$$

где  $R$  – матрица вращения,  $p$  – вектор переноса.

Кинематика конечности определяется на основе подхода «Произведение экспонент» (Product of Exponentials, PoE) и принимает вид:

$$T(q) = e^{[S_1]\theta_1} e^{[S_2]\theta_2} \dots e^{[S_n]\theta_n} M, \quad (16)$$

где  $S_i$  – винтовая ось  $i$ -ого сустава (Twist  $S$ ),  $\theta_i$  – угол  $i$ -ого сустава,  $M$  – поза конечности при нулевых углах.

Производится определение физических критериев устойчивости, которые заключаются в расположении координат точек центра масс и нулевого момента в области опорного многоугольника, образованного опорными конечностями робота. Применение концепции критических состояний, предлагает включить их в состав функции близости к неустойчивости для случая эвристического представления. Тогда формализация задачи управления локомоцией средствами оптимизации с учётом критических состояний, примет вид:

$$\begin{aligned} J(\theta) &= E_{\tau \sim \pi_\theta} [R(\tau)] \rightarrow \max \\ E_{\pi_\theta} [C(s_i)] &\leq \delta \\ \varepsilon_{\theta_i} &\leq \Theta_i \leq \varepsilon'_{\theta_i} \\ \eta'_i &\leq \eta \leq \eta''_i \\ \mu'_i &\leq \mu \leq \mu''_i \\ \sigma'_i &\leq \sigma \leq \sigma''_i \\ l &\in L \end{aligned} \quad (17)$$

где  $\varepsilon_{\theta_i}$  и  $\varepsilon'_{\theta_i}$  – нижняя и верхняя границы рабочего диапазона углов сочленений,  $\eta'_i$  и  $\eta''_i$  – нижняя и верхняя границы рабочего диапазона углов тангажа для локомоторной программы  $l$ ,  $\mu'_i$  и  $\mu''_i$  – крена,  $\sigma'_i$  и  $\sigma''_i$  – рысканья,  $L$  – множество всех локомоторных команд.

Таким образом, было получено математическое обеспечение системы управления интеллектуальным агентом, отличающееся внедрением в агента возможности предсказания риска возникновения нестабильных состояний, основанной на выделенном множестве критических состояний, и обеспечивающее повышение безопасности обучения и стабилизацию функционирования управляемой киберфизической системы.

**Третья глава** посвящена построению алгоритмического обеспечения системы управления интеллектуальным агентом, обеспечивающего минимизацию коллизий в процессе обучения и многофункциональность агента на уровне глубокой нейронной сети. В начале, производится структурное разделение системы управления на основные функциональные блоки: агента – блок принятия решений и среду – блок, отвечающий за считывание текущего наблюдения, генерацию сигнала завершения эпизода и формирование вознаграждения. Структурная схема системы управления и схема взаимодействия внутренних блоков высокоуровневых структурных элементов представлена на рис. 2.

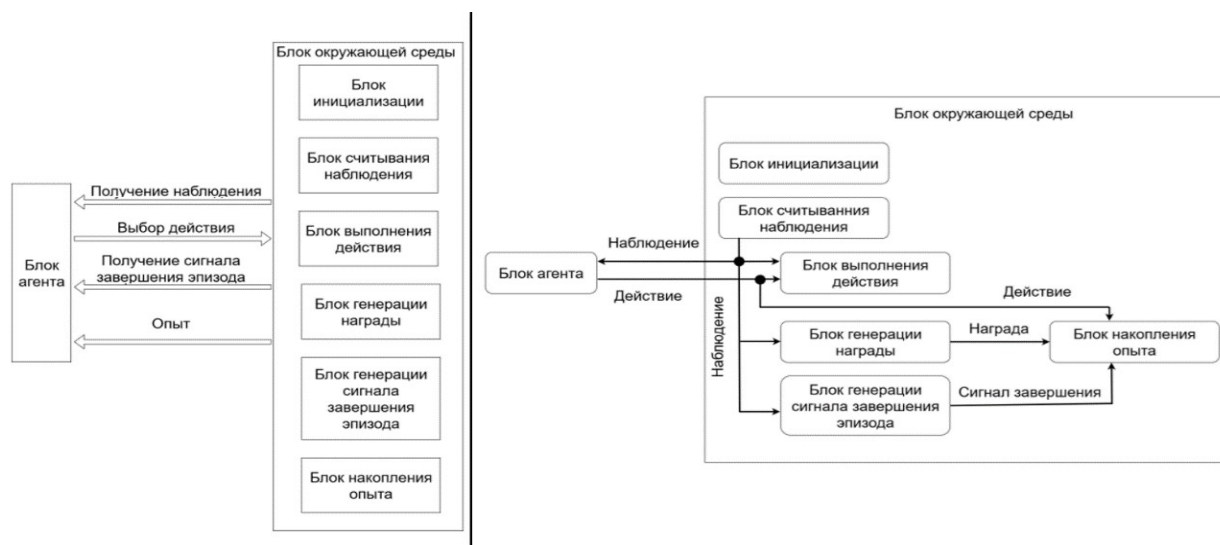


Рис. 2 – Структурная схема системы управления (слева) и схема взаимодействия внутренних блоков структурных элементов (справа)

Представление функции рисков  $C(s)$  в эвристическом виде обладает множеством ограничений. Агент не видит сложную динамику (периодически не улавливает будущее падение). Требуется ручная настройка параметров и не переносится между разными системами. Поэтому реализуется обучаемая модель риска  $C_\phi(s)$ , аппроксимирующая вероятность попадания агента в нестабильное состояние через  $k$  шагов.

Предлагается алгоритм ликвидации критических состояний, обеспечивающий минимизацию коллизий, путём учёта соответствующих состояний, и делающий акцент обучения на границе устойчивости. Агент учится не просто не попадать в неустойчивые состояния, а вообще не приближаться к ним. Другими словами, в обучение с подкреплением концептуально вводится «осознание риска», в результате чего агент, чувствуя приближение неустойчивого состояния, учится избегать его.

Предлагаемый алгоритм ликвидации критических состояний имеет следующий вид:

*Начало алгоритма*  
*Инициализировать стратегию  $\pi$*   
*Инициализировать value-функцию  $V$*   
*Инициализировать модель риска  $C_\phi$*   
*Инициализировать множитель  $\lambda$*   
*Задать параметры  $\alpha, \beta, \delta, C_{crit}, C_{max}, k$*   
*Пока не выполнен критерий останова обучения:*  
    *Начало эпизода*  
    *Если есть сохранённое критическое состояние:*  
        *Установить начальное состояние  $s$  из буфера (откат на  $k$  шагов по траектории  $\tau$ )*  
    *Иначе:*  
        *Инициализировать  $s$  стандартным образом*  
        *Сбросить флаг  $critical\_flag = ЛОЖЬ$*   
    *Для каждого шага эпизода:*  
        *Выбрать действие  $a$  согласно стратегии  $\pi$  в состоянии  $s$*   
        *Выполнить действие  $a$*   
        *Получить новое состояние  $s_{next}$  и награду  $r$*   
        *Вычислить предсказанный риск  $C_{pred}$  с помощью модели  $C_\phi$*   
        *Вычислить модифицированную награду:*  
            
$$r_{mod} = r - \lambda \cdot C_{pred}$$
  
        *Если произошло попадание в неустойчивое состояние или  $C_{pred} \geq C_{max}$  :*  
            *Установить флаг окончания эпизода  $done$  в ИСТИНА*  
        *Иначе:*  
            *Установить флаг окончания эпизода  $done$  в ЛОЖЬ*  
        *Если  $C \geq C_{crit}$  :*  
            *Установить флаг наличия критического состояния  $critical\_flag$  в ИСТИНА*  
        *Сохранить переход  $(s, a, r_{mod}, C_{pred}, done)$  в буфер*  
        *Если флаг окончания эпизода  $done$  в значении ИСТИНА:*  
            *Прервать цикл шагов*  
        *Обновить состояние:*  
            
$$s = s_{next}$$
  
    *Конец цикла шагов*  
    *Начало обновления моделей*  
    *Вычислить преимущества  $advantage$  и обесцененные доходы  $return$  по буферу*  
    *Обновить параметры стратегии  $\pi$  и value-функции  $V$  (алгоритм PPO)*

Для каждого сохранённого состояния:

Определить метку  $y$ :

$y=1$ , если в течение следующих  $k$  шагов было попадание в нестабильное состояние

$y=0$  иначе

Обновить параметры модели риска  $S_f$  по ошибке предсказания

Вычислить средний риск  $C_{mean}$  по эпизоду

Обновить множитель  $\lambda$ :

$$\lambda = \max(0, \lambda + \beta \cdot (C_{mean} - \delta))$$

Конец обновления моделей

Начало процедуры сброса

Если  $critical\_flag = ИСТИНА$ :

Выбрать состояние из буфера с откатом на  $k$  шагов

Использовать его как начальное для следующего эпизода

Иначе:

Использовать стандартную инициализацию

Очистить буфер

Конец процедуры сброса

Конец цикла обучения

Конец алгоритма

Таким образом, было получено алгоритмическое обеспечение, позволяющее агенту безопасно обучаться на взаимодействии с окружающей средой посредством ликвидации критических состояний и обеспечивать более безопасный перенос агента с последующей безопасной настройкой в реальной среде.

Далее рассматривается проблематика многофункциональности агента. В работе предлагается новая архитектура нейронной сети, обеспечивающая решение проблемы низкой репрезентативной способности глубокой сети. Вводится определение нейронного кластера - совокупность нейронов, полностью связанных между собой. Условно разделим кластер на несколько частей (нейронные группы). Количество нейронных групп внутри нейронного кластера характеризует его *программную ёмкость*  $W$  (количество программ, которое способен аппроксимировать нейронный кластер). Каждая группа ассоциируется с конкретной решаемой задачей, тем самым обеспечивая выполнение принципа «одна задача – одна сеть», исключая проблему забывания. Для осуществления селекции групп предлагается использовать уникальные бинарные коды, соответствующие каждой из групп. Пусть задана некоторая функция  $\Psi(p)$ , ставящая в соответствие коду стратегии  $p$  некоторую битовую маску  $m$ , активирующую нейроны выбранной группы и деактивирующая все остальные. Эта функция в нейронной сети будет являться переключающим нейроном – искусственным нейроном базального ганглия, ставящим в соответствие коду стратегии  $p$  (равносильно движению) некоторую

активационную маску  $m$  группы внутри кластера (эквивалент кодировки активации, генерируемой нейроном базального ганглия).

Предложенная архитектура выполняет разделение полносвязной нейронной сети между задачами, обеспечивая их независимое запоминание без «потери памяти» при переключении агента между ними. Каждый искусственный нейрон базального ганглия отвечает за активацию своей группы нейронов в слое, не пересекаясь с нейронами других групп. Предлагаемый механизм переключающих нейронов может быть интерпретирован как миварный механизм выбора стратегии, при котором код стратегии играет роль правила вывода, а нейронная группа выступает реализацией соответствующего сценария управления. Ограничение прогрессивной нейронной сети, путём введения программной ёмкости  $W$ , что означает, что сеть содержит  $W$  столбцов, и добавление переключающего нейрона на каждый глубокий слой нейросети, включая выходной, позволяют полноценно контролировать селекцию стратегий, путём передачи соответствующего стратегии кода. В результате получается следующая прогрессивная сеть, управляемая переключающими нейронами в контексте нейронного кластера, представленная на рис. 3.

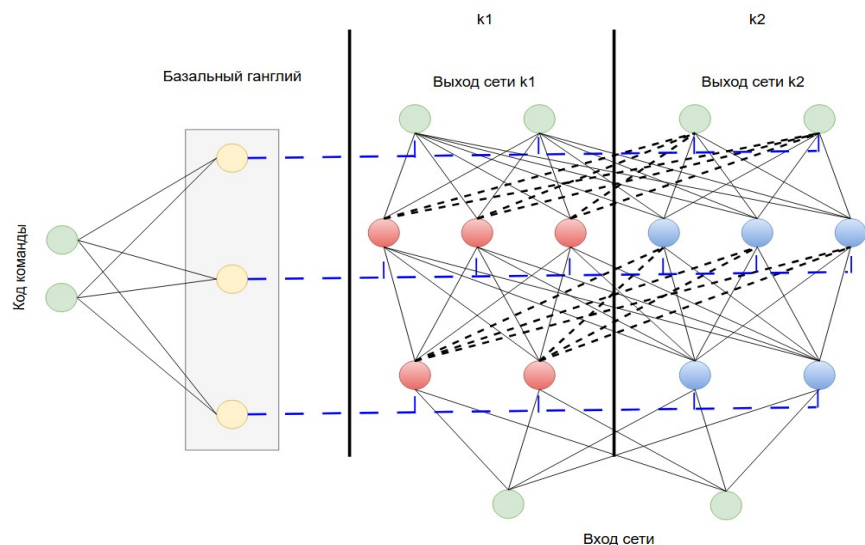


Рис. 3 – Прогрессивная нейронная сеть на основе полносвязного нейронного кластера с использованием переключающих нейронов

Таким образом, использование предлагаемой архитектуры к прогрессивным нейронным сетям видоизменяет процесс предсказания выхода. Поскольку кластер – это полносвязная сеть, то следует брать в рассмотрение исключительно «правые» соединения (от левого столбца к правому). Ликвидация «левых связей» производится на основе имеющихся активационных масок управляющих нейронов и дополнительных вычислений, производимых на этапе предсказания.

Изменённый алгоритм предсказания выхода, в контексте прогрессивных нейронных сетей с использованием переключающих нейронов, имеет следующий вид:

*Шаг 1: Выполнить расчёт выходов  $y_1$  слоя  $h_1$ .*

*Шаг 2: Получить маску переключающего нейрона для слоя  $h_2$ .*

*Шаг 3: Согласно полученной маске для слоя  $h_2$  вычислить выходные значения для активных нейронов (вектор значений  $y_2^1$ ) слоя  $h_2$  по всем связям с нейронами слоя  $h_1$ .*

*Шаг 4: Выполнить инверсию маски, полученной от переключающего нейрона для слоя  $h_2$ .*

*Шаг 5: Получить маску переключающего нейрона для слоя  $h_1$ .*

*Шаг 6: Выполнить инверсию маски, полученной от переключающего нейрона для слоя  $h_1$ .*

*Шаг 7: На основе инвертированных масок, полученных на шагах 4 и 6 вычислить выходные значения для активных нейронов слоёв  $h_1$  и  $h_2$  – вектор значений  $y_2^2$ .*

*Шаг 8: Сформировать рассчитанный вектор выходов слоя  $h_2$  по следующей формуле:  $y_2 = \text{concat}(y_2^1, y_2^2)$ , где  $\text{concat}$  – операция слияния векторов в один.*

*Шаг 9: Получить маску переключающего нейрона для слоя выходов.*

*Шаг 10: Согласно полученной маске для слоя  $h_2$  вычислить выходные значения  $y_3$  для активных нейронов выходного слоя по всем связям с нейронами слоя  $h_2$ .*

Далее приводятся рекомендации по формированию нейронного кластера. Таким образом, была разработана архитектура глубокой нейронной сети, обеспечивающая агента, основанного на глубоком обучении с подкреплением, возможностью аппроксимации нескольких стратегий в контексте одной глубокой нейронной сети и ликвидирующей эффект «забывания» и потерю деталей каждой из стратегий.

**Четвертая глава** посвящена построению структуры системы управления интеллектуальным агентом, обеспечивающей многофункциональность и ликвидацию коллизий в процессе обучения на примере киберфизической системы шагающего робота, состоящего из двух педипуляторов, каждый из которых обладает пятью степенями свободы. Рассматривается функция вознаграждения для прямолинейного движения, которая имеет следующий вид:

$$R = \alpha_{fwd} \cdot v_x + \alpha_{dir} \cdot e^{-k_\theta |\theta|} - \alpha_{stab} \cdot (|\theta_{roll}| + |\theta_{pitch}|) - \alpha_\omega \cdot \|\omega\|^2 - \alpha_{cont} \cdot (\|v_{foot}\| \cdot 1_{cont}) - \lambda \cdot C(s) \quad (18)$$

где  $v_x$  – скорость при движении вперёд,  $\theta$  – угол отклонения от направления вперёд,  $k_\theta, \alpha_\omega$  – некоторые настраиваемые коэффициенты,  $\theta_{roll}$  – угол крена,  $\theta_{pitch}$  – угол тангажа,  $\omega$  – угловая скорость,  $v_{foot}$  – скорость конечности,  $1_{cont}$  – индикаторная функция (при касании равна 1, иначе 0),

$\alpha_{fwd}, \alpha_{dir}, \alpha_{stab}, \alpha_{cont}$  – весовые коэффициенты,  $\lambda$  – параметр Лагранжиана,  $C(s)$  – функция расстояния до множества нестабильных состояний.

Производится обоснование выбора глубокого алгоритма обучения с подкреплением PPO, на основании полученных, в контексте исследования, экспериментальных данных. Далее реализуется построение агента. Конструирование производится с учётом концепции критических состояний. В результате, структура системы управления имеет вид, представленный на рис. 4.

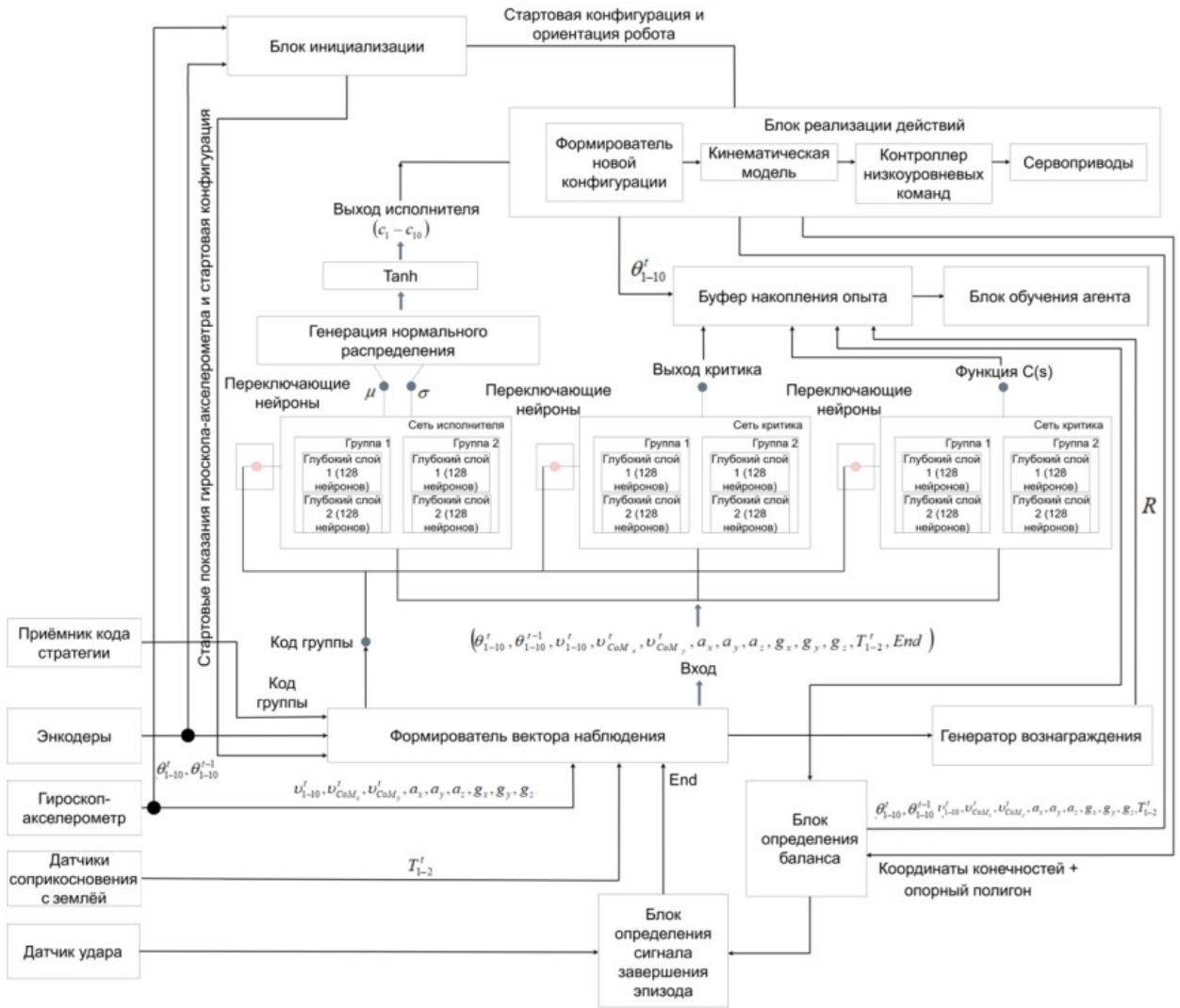


Рис. 4 – Структура системы управления

Для обоснования эффективности предложенной структуры системы управления, ее математического и алгоритмического обеспечения было проведено множество экспериментов. Апробация предложенного в контексте исследования алгоритма ликвидации критических состояний производилась в сравнении с базовым алгоритмом, не учитывающим соответствующие состояния в процессе функционирования. Экспериментальные данные были

получены на 6000 эпизодах. Длина эпизода соответствовала 300 шагов. Параметр скорости обучения сетей исполнителя и критика соответствовал 0,0001. Количество эпох на итерацию равнялось 3.

В качестве метрик качества для реализации сравнения были выбраны: *процент эпизодов без фактического падения, процент полностью завершённых эпизодов, максимальная длина эпизода, количество шагов до падения и доля случаев, когда робот был в падении, но восстановился*. В результате экспериментов были получены следующие результаты, представленные в таблице 1.

Таблица 1

Полученные показатели метрик качества

Метрика	Классический	Предлагаемый
Процент эпизодов без фактического падения	65%	85%
Процент полностью завершённых эпизодов	65%	80%
Максимальная длина эпизода	300	300
Количество шагов до падения	300	1000
Доля случаев, когда робот был в падении, но восстановился	5%	55%

Апробация архитектуры с переключающимися нейронами проводилась для проверки гипотезы о том, что разные группы реализуют разные режимы локомоции и могут стабильно переключаться. Для эксперимента использовалась одна и та же локомоция — движение прямо — на двух типах поверхности: ровной и песчаной. Метрикой служил *процент эпизодов без падения*. Сначала обучалась каждая стратегия, затем измерялось число успешных эпизодов в среде, соответствующей её коду. Эти результаты принимались за эталон. Предполагалось, что при смене условий на противоположные показатель заметно изменится, если переключение и аппроксимация стратегий выполняются корректно. Экспериментальные данные приведены в таблице 2.

Таблица 2

Метрика количества падений для комбинаций стратегий

Стратегия обучения	Переход	Процент эпизодов без падения
Прямо по ровной поверхности	Ровная - Ровная	85
Прямо по песчаной поверхности	Песчаная - Песчаная	80
Прямо по ровной поверхности	Ровная - Песчаная	40
Прямо по песчаной поверхности	Песчаная - Ровная	85

Из полученных результатов видно, что при переходе обученной стратегии движения по ровной поверхности на песчаную, заметно

значительное снижение количества эпизодов без падений (более чем в 2 раза). Это связано с тем, что данная стратегия не учитывает проскальзывание и делает агрессивный шаг. В отличие от неё, стратегия обученная движению по песку более осторожна и учитывает вязкость. Поэтому при переходе на ровную поверхность заметно возрастание процента эпизодов без падения. Таким образом, каждая группа корректно аппроксимирует соответствующую ей стратегию. Выполняется корректная селекция стратегий.

В результате была разработана структура системы управления интеллектуальным агентом, обеспечивающая автономное адаптивное безопасное обучение множеству стратегий с последующей безопасной настройкой на примере управления шагающей локомоцией робота.

### **ЗАКЛЮЧЕНИЕ**

В процессе выполнения диссертационного исследования были получены следующие основные результаты:

1. Выполнен анализ существующих подходов к построению систем управления интеллектуальным агентом на основе глубокого обучения с подкреплением, включая проблематику управления локомоцией шагающих роботов, как частного случая киберфизических систем, в контексте поиска наиболее подходящего для реализации цели построения адаптивного автономного управления.

2. Разработано специальное математическое обеспечение системы управления интеллектуальным агентом, обеспечивающее повышение уровня безопасности и устойчивости функционирования управляемой киберфизической системы.

3. Разработано специальное алгоритмическое обеспечение системы управления интеллектуальным агентом, обеспечивающее ускорение обучения и переноса агента на реальную систему за счёт увеличения времени устойчивого функционирования и сокращения числа нештатных режимов (коллизий).

4. Разработана архитектура глубокой нейронной сети, обеспечивающая агента, основанного на глубоком обучении с подкреплением, возможностью обучения нескольким стратегиям в контексте одной глубокой нейронной сети и ликвидирующей эффект «забывания» и потерю деталей каждой из стратегий.

5. Разработана структура системы управления интеллектуальным агентом, обеспечивающая автономное адаптивное безопасное обучение множеству стратегий с последующим переносом на реальную киберфизическую систему и безопасную настройку на примере формирования локомоции шагающего робота.

6. Разработанные элементы программного обеспечения системы управления локомоцией мобильного шагающего робота зарегистрированы в ФИПС.

7. Выполнен цикл экспериментов по определению эффективности разработанного математического и алгоритмического обеспечения

безопасного обучения агента и предложенной архитектуры глубокой нейронной сети на примере решения задачи управления локомоцией киберфизической системы мобильного шагающего робота. Результаты экспериментов продемонстрировали увеличение устойчивости агента на 20%, при использовании алгоритма ликвидации критических состояний в сравнении с базовым алгоритмом, не учитывающим соответствующие состояния. Так же наблюдался значительный прирост значения метрики «количество шагов до падения» и «доля случаев, когда робот был в падении, но восстановился».

### **Рекомендации и перспективы дальнейшей разработки темы**

1. Результаты исследования целесообразно применять при проектировании адаптивных автономных систем управления, использующих алгоритмы обучения с подкреплением, основанные на глубоких нейронных сетях. Предложенная архитектура многофункционального агента и концепция критических состояний применимы для решения задач управления в областях интеллектуальной робототехники и построения систем искусственного интеллекта.

2. Дальнейшую разработку темы следует направить на изучение применения концепции нейронного кластера к другим типам глубоких сетей, таким как рекуррентные нейронные сети и сети с памятью (LSTM, GRM). Также следует рассмотреть задачу классификации окружающей среды для построения высокоуровневого генератора кодов селектора стратегий.

### **Основные результаты диссертации опубликованы в следующих работах:**

#### **Публикации в изданиях списка ВАК**

1. Кашко В.В., Олейникова С.А. Математическая модель универсальной системы управления шагающим роботом на основе методов обучения с подкреплением. Моделирование, оптимизация и информационные технологии. 2024;12(1). URL: <https://moitvvt.ru/ru/journal/pdf?id=1520> DOI: 10.26102/2310-6018/2024.44.1.025.

2. Кашко, В. В. Анализ применения параметрических функций вознаграждения в алгоритмах обучения с подкреплением / В. В. Кашко, С. А. Олейникова, Ю. С. Акинина // Вестник Воронежского государственного технического университета. — 2024. — Т. 20, № 3. — С. 37–43. — DOI: 10.36622/1729-6501.2024.20.3.005.

3. Кашко, В. В. Анализ сходимости и устойчивости алгоритма глубокого обучения с подкреплением исполнитель критик A2C / В. В. Кашко, С. А. Олейникова // Вестник Воронежского государственного технического университета. — 2024. — Т. 21, № 1. — С. 12–19. — DOI: 10.36622/1729-6501.2025.21.1.002.

4. Кашко, В. В. Общий алгоритм ликвидации критических состояний для решения задачи управления реальным шагающим роботом на основе методов глубокого обучения с подкреплением / В. В. Кашко, С. А.

Олейникова // Программные системы и вычислительные методы. — Москва, 2025. — № 3. — С. 103–114. — DOI: 10.7256/2454-0714.2025.3.75996.

5. Кашко, В. В. Сравнительный анализ алгоритмов A2C и PPO с различными архитектурами нейронных сетей на примере среды CartPole / В. В. Кашко, С. А. Олейникова // Вестник Воронежского государственного университета. — 2026. — Т. 20, № 3. — С. 37–43.

6. Кашко, В. В. Архитектура нейронной сети многофункционального агента на базе глубокого обучения с подкреплением / В. В. Кашко, С. А. Олейникова // Вестник Воронежского государственного технического университета. — 2026. — Т. 22, № 1. — С. 35–44. — DOI: 10.36622/1729-6501.2026.22.1.005.

#### **Свидетельства о государственной регистрации программ для ЭВМ**

7. Гребенникова Н.И., Олейникова С.А., Кашко В.В. Интеллектуальная система поддержки принятия решений на базе аппарата искусственных нейронных сетей // Свидетельство о регистрации программы для ЭВМ № 2023665763 от 02.07.2023, г. Москва, Федеральный институт промышленной собственности.

#### **Статьи в журналах, сборниках трудов конференций**

8. Кашко, В. В. Анализ проблем обучения нейронных сетей и методов их устранения / В. В. Кашко, С. А. Олейникова // Информационные технологии моделирования и управления. — 2023. — Т. 132, № 2. — С. 124–128.

9. Кашко, В. В. Применение методов обучения с подкреплением для реализации движения шагающих роботов / В. В. Кашко, С. А. Олейникова // Современные информационные технологии. Теория и практика. — 2024. — С. 256–262.

10. Кашко, В. В. Анализ методов обучения с подкреплением для управления роботизированными системами / В. В. Кашко, С. А. Олейникова // Инновационные технологии: теория, инструменты, практика. — 2024. — Т. 1. — С. 133–140.

11. Kashko, V. V. Formalization of the task of controlling the movement of a walking robot / V. V. Kashko, S. A. Oleinikova // Anthropocentric sciences in education: challenges, transformations, resources. — 2024. — P. 342–345.

12. Кашко, В. В. Влияние Dropout-регуляризации глубокой сети на сходимость и устойчивость алгоритма градиента стратегии Advantage Actor Critic (A2C) / В. В. Кашко, С. А. Олейникова // Вестник Пензенского государственного университета. — 2025. — № 1(49). — С. 62–65.

13. Кашко, В. В. Влияние архитектуры глубокой нейронной сети на функционирование алгоритма градиента стратегии исполнитель-критик A2C / В. В. Кашко, С. А. Олейникова // Актуальные проблемы прикладной математики, информатики и механики: сборник трудов Международной научной конференции. — 2025. — С. 233–238.

14. Кашко, В. В. Анализ качества решения задачи управления интеллектуальным агентом на основе алгоритма градиента стратегии

REINFORCE с точки зрения особенностей окружающей среды / В. В. Кашко // Управление программным инжинирингом: труды открытой Международной научно-практической конференции. — Воронеж, 2025. — С. 44–49.

15. Кашко, В. В. Формализация задачи управления шагающим роботом на основе алгоритмов обучения с подкреплением / В. В. Кашко, С. А. Олейникова // Интеллектуальные информационные системы: труды Международной научно-практической конференции. — Воронеж, 2025. — С. 243–247.

16. Кашко, В. В. Обобщённый алгоритм решения задачи управления шагающим роботом на базе интеллектуального агента с использованием методов глубокого обучения с подкреплением / В. В. Кашко, С. А. Олейникова // Научная опора Воронежской области: сборник трудов победителей конкурса научно-исследовательских работ студентов и аспирантов ВГТУ по приоритетным направлениям развития науки и технологий. — Воронеж, 2025. — С. 155–158.

17. Кашко, В. В. Разработка подхода для сопряжения нейронных групп внутри полносвязной сети-кластера с использованием нейронов-повторителей / В. В. Кашко, С. А. Олейникова // Инновационные технологии: теория, инструменты, практика: материалы XVII Международной интернет-конференции молодых учёных, аспирантов, студентов (17 ноября – 30 декабря 2025 г.) / Пермский национальный исследовательский политехнический университет. — Пермь, 2025.

18. Кашко, В. В. Построение многофункционального агента на базе глубокого обучения с подкреплением / В. В. Кашко, С. А. Олейникова // Актуальные проблемы прикладной математики, информатики и механики: сборник трудов Международной научной конференции. — 2025. — С. 285–291.

19. Кашко, В. В. Экспериментальное обоснование эффективности разработанной системы управления шагающим роботом / В. В. Кашко, С. А. Олейникова // Информационные технологии в строительных, социальных и экономических системах. — Воронеж, 2026. — С. 70–75.

Подписано в печать 29.06.2026.

Формат 60x84/16. Бумага для множительных аппаратов.

Усл. печ. л. 1,0. Тираж 80 экз. Заказ № 115

ФГБОУ ВО «Воронежский государственный технический университет»

394006 Воронеж, 20-летия Октября, 84