ISSN 2658-5014 (Print) СИИТ ISSN 2686-7044 (Online)
http://siit.ugatu.su 2025. Т. 7, № 5(24)
СИСТЕМНАЯ ИНЖЕНЕРИЯ И ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ
Научная статья

УДК 004.8 DOI: 10.54708/2658-5014-SIIT-2025-no5-p86 EDN: AMLCRV

Сравнение моделей нейронных сетей для автоматического управления полетом квадрокоптера по заданной траектории

Р. Д. Халилов • Т. З. Муслимов

Пропорционально-интегрально-дифференциальные (ПИД) регуляторы широко применяются в промышленности и исследовательских задачах благодаря простоте и эффективности. Однако при наличии параметрических неопределенностей и внешних возмущений, особенно в динамически сложных системах вроде квадрокоптеров, остаётся актуальной задача обеспечения их робастности. В работе сравнивается самонастраивающаяся ПИД-схема, использующая подкрепляющее обучение и гибридную нейросетевую архитектуру «актор–критик» для управления ориентацией и высотой полёта квадрокоптера без априорной математической модели, с подобной архитектурой, использующей метод Proximal Policy Optimization (PPO) для оптимизации работы. В обоих случаях коэффициенты усиления регулятора состоят из статической и адаптивной динамической части, при этом обучаются только переменные компоненты. Нейросеть включает два скрытых слоя с сигмоидальными активациями. Обучение проводилось онлайн с оптимизатором ADAM и обратным распространением ошибки, что обеспечивает быструю адаптацию ко внешним возмущениям и изменению массы аппарата. Эксперименты показали высокую устойчивость систем к вариациям массы и порывам ветра при использовании траекторий различной сложности. Сравнение двух методов показало, что значительной разницы в отклонениях от идеальной траектории у них нет, однако метод PPO обучался в 2.8 раза быстрее, чем стандартный «актор–критик». Кроме того, метод PPO показал большее отклонение от идеальной высоты при изменении массы дрона в полёте. Результаты подтверждают потенциал гибридных нейросетевых структур для адаптивного управления в условиях неопределённости и рекомендуют разработанный алгоритм к практическому применению в автономных БПЛА, при этом архитектура, использующая стандартную модель «актор–критик», предпочтительнее при изменениях массы квадрокоптера в полёте, а архитектура, использующая PPO – при сложных, длинных маршрутах

Адаптивное ПИД-регулирование; обучение с подкреплением; квадрокоптер; нейросеть; «актор–критик»; самонастраивающийся регулятор; БПЛА.

Введение

Актуальность исследования обусловлена стремительным ростом применения беспилотных летательных аппаратов (БПЛА) в различных сферах человеческой деятельности. Квадрокоптеры находят применение в мониторинге промышленных объектов [Sha20, Гур24], доставке грузов [Fan19], поисково-спасательных операциях [Wah10] и сельском хозяйстве [Gup25, Сан25]. Однако эффективное выполнение этих задач требует высокоточной стабилизации и точного следования сложным траекториям в условиях внешних возмущений и параметрических неопределенностей.

Проблематика управления квадрокоптерами связана со свойственной им нелинейной динамикой, сильной взаимосвязью каналов управления и чувствительностью ко внешним воздействиям, таким как порывы ветра [Ban16]. Традиционные ПИД-регуляторы, несмотря

Халилов Р. Д., Муслимов Т. З. Сравнение моделей нейронных сетей для автоматического управления полетом квадрокоптера по заданной траектории // СИИТ. 2025. Т. 7, № 5(24). С. 86-108. DOI: 10.54708/2658-5014-SIIT-2025-no5-p86. EDN: AMLCRV.

Khalilov R. D., Muslimov T. Z. Comparison of neural network models for automatic flight control of a quadcopter along a given trajectory // SIIT. 2025. Vol. 7, no. 5(24), pp. 86-108. DOI: 10.54708/2658-5014-SIIT-2025-no5-p86. EDN: AMLCRV (In Russian).

на свою простоту и широкое распространение [Åst22], демонстрируют ограниченную эффективность в условиях изменяющихся динамических характеристик и внешних возмущений [Lop23].

Обзор современных подходов показывает разнообразие методов решения задачи управления мобильными роботами различных типов [Гур24, Мус24], в число которых входят и квадрокоптеры. Li и др. [LiY24] предложили комбинацию управления с прогнозирующей моделью (Model Predictive Control – MPC) с нейронными сетями для трекинга траекторий, однако их подход требует знания точной математической модели. В работе [Ngu21] исследуется адаптивное управление на основе нейросетей, но с обучением «оффлайн» (off-line), что ограничивает применимость в реальных условиях. Методы глубокого обучения с подкреплением [Sut18] демонстрируют перспективность, но часто требуют значительных вычислительных ресурсов.

В данной работе сравнивается подход, предложенный в [Ima22], с аналогичным подходом, но использующим модель Proximal Policy Optimization (PPO) [Sch17] в различных условиях. В качестве первого подхода используется метод «актор–критик» (Actor-Critic – A2C), который зарекомендовал себя как эффективный подход для задач управления с непрерывным пространством действий [Gro12]. В контексте управления квадрокоптерами данный метод позволяет сочетать превосходство градиента с функцией ценности состояния [Tan18]. Однако существующие реализации часто страдают от медленной сходимости и чувствительности к гиперпараметрам [Hen22]. В качестве второго метода используется Proximal Policy Optimization (PPO), который также доказал свою эффективность при работе с квадрокоптерами [Zha24].

Оптимизатор ADAM [Kin15] стал де-факто стандартом для обучения глубоких нейросетей благодаря адаптивной настройке скорости обучения и устойчивости к шуму в градиентах. В сочетании с алгоритмом обратного распространения ошибки [Rum86] он обеспечивает эффективную оптимизацию даже для невыпуклых функций потерь, характерных для задач обучения с подкреплением.

Пробел в исследованиях заключается в отсутствии эффективных методов онлайн-адаптации, сочетающих надежность традиционных регуляторов с возможностью обучения с помощью нейросетевых подходов.

Большинство существующих решений либо требуют точной модели объекта, либо осуществляют обучение до развертывания системы, что ограничивает их применимость в условиях непредсказуемых изменений динамики [Саи25, При25].

Основной вклад данной работы заключается в:

- проверке стабильности работы гибридной архитектуры «актор–критик» (Actor-Critic – A2C), сочетающей ПИД-регуляторы с нейросетевой адаптацией коэффициентов;
- разработке структуры PPO, сочетающей ПИД-регуляторы с нейросетевой адаптацией коэффициентов;
- сравнении устойчивости и эффективности работы обоих подходов;
- экспериментальной валидации устойчивости к параметрическим неопределенностям и внешним возмущениям;
- создании симуляционной модели в робототехническом симуляторе CoppeliaSim для верификации подхода.

Практическая значимость исследования состоит в потенциальном применении алгоритма в:

- промышленном мониторинге (инспекция трубопроводов, ЛЭП);
- поисково-спасательных операциях в сложных погодных условиях;
- сельскохозяйственных технологиях;
- доставке медицинских грузов в удаленные районы.

Статья организована следующим образом: описана математическая модель квадрокоптера; детализирована архитектура системы управления; представлены экспериментальные результаты; сформулированы выводы и направления будущих исследований.

Математические модели системы управления

Квадрокоптер (рис. 1) представляет собой управляемую систему с четырьмя управляющими воздействиями (u_1, u_2, u_3, u_4) и шестью степенями свободы: тремя координатами положения (x, y, z) и тремя углами ориентации (ϕ, θ, ψ). Из-за выраженных нелинейностей модели и влияния нестационарных внешних факторов точное математическое описание аппарата затруднительно. В подобных условиях методы идентификации, в частности на основе нейронных сетей, позволяют адекватно восстанавливать текущее состояние системы, что исключает необходимость в детальной модели – достаточно лишь мгновенных входных и выходных данных.

В работе применяется простая модель [Tri15] с известными параметрами для воспроизведения поведения реального объекта без шума. При этом параметры изначально считаются неизвестными, а состояния оцениваются на основе управляющих сигналов и последующих наблюдений. Математическое описание динамики задано в уравнениях (1), где x, y, z – координаты центра масс относительно системы координат x_I, y_I, z_I (для осей этой системы координат далее также применяется обозначение *X, Y, Z*), а ϕ, θ, ψ – углы вращения вокруг осей x_B, y_B, z_B (см. рис. 1).

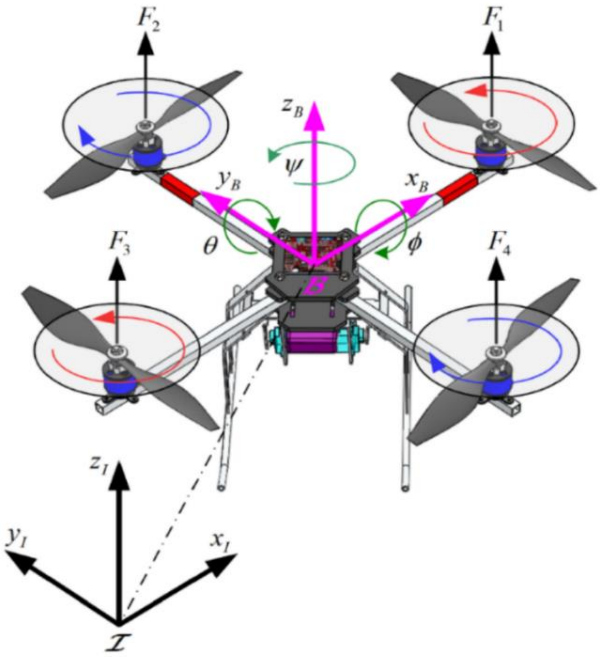

Рис. 1 Схема квадрокоптера [Ima22]

$$\ddot{\phi} = \dot{\theta}\dot{\psi}\frac{J_y - J_z}{J_x} + \frac{l}{J_x}u_2, \quad \ddot{\theta} = \dot{\phi}\dot{\psi}\frac{J_z - J_x}{J_y} + \frac{l}{J_y}u_3, \quad \ddot{\psi} = \dot{\phi}\dot{\theta}\frac{J_x - J_y}{J_z} + \frac{1}{J_z}u_4,$$
$$\ddot{x} = \frac{u_1}{m}(\cos\phi\sin\theta\cos\psi + \sin\phi\sin\psi), \ddot{y} = \frac{u_1}{m}(\cos\phi\sin\theta\sin\psi - \sin\phi\cos\psi), \qquad (1)$$
$$\ddot{z} = \frac{u_1}{m}\cos\phi\sin\theta - g,$$

где m, g, l – масса дрона, ускорение свободного падения и длина плеча; J_x, J_y, J_z – моменты инерции относительно соответствующих осей.

Управляющие воздействия являются комбинацией угловых скоростей вращения пропеллеров ($\Omega_1, \Omega_2, \Omega_3, \Omega_4$):

$$u_1 = b(\Omega_1^2 + \Omega_2^2 + \Omega_3^2 + \Omega_4^2), \quad u_2 = b(\Omega_4^2 - \Omega_2^2),$$
$$u_3 = b(\Omega_3^2 - \Omega_1^2), \quad u_4 = d(\Omega_4^2 + \Omega_2^2 - \Omega_1^2 + \Omega_3^2), \qquad (2)$$

где b и d – коэффициенты тяги и крутящего момента соответственно.

Также необходимо определить управляющие сигналы с использованием ПИД-регулятора в онлайн-режиме. Сначала статические коэффициенты подбираются экспериментально или по методу Циглера–Никольса до обеспечения устойчивости и приемлемого качества; эти значения фиксируются для каждого этапа. Динамические коэффициенты рассчитываются данным алгоритмом и добавляются к статическим.

$$u(t) = K_p(t)e(t) + K_i(t)\int_0^t e(\tau)d\tau + K_d(t)\frac{de(t)}{dt}. \quad (3)$$

Коэффициенты усиления формируются как сумма статической и динамической составляющих:

$$\begin{bmatrix} K_p(t) \\ K_i(t) \\ K_d(t) \end{bmatrix} = \begin{bmatrix} K_p{}^{\text{static}}(t) \\ K_i{}^{\text{static}}(t) \\ K_d{}^{\text{static}}(t) \end{bmatrix} + \Delta \begin{bmatrix} K_p{}^{\text{dynamic}}(t) \\ K_i{}^{\text{dynamic}}(t) \\ K_d{}^{\text{dynamic}}(t) \end{bmatrix} \quad (4)$$

где $K_p{}^{\text{static}}$, $K_i{}^{\text{static}}$, $K_d{}^{\text{static}}$ – постоянные, заранее определенные величины, а $K_p{}^{\text{dynamic}}$, $K_i{}^{\text{dynamic}}$, $K_d{}^{\text{dynamic}}$ – величины, динамически генерируемые нейросетевой моделью на основе текущего состояния системы. Нейросеть на основе метода «актор–критик» подробно рассмотрена ниже.

При построении модели приняты следующие допущения:

- квадрокоптер рассматривается как симметричное твердое тело;
- влияние гироскопических эффектов двигателей учитывается через J_i;
- аэродинамические силы сопротивления пропорциональны квадрату скорости;
- учитываются ограничения на углы крена и тангажа: $|\phi|, |\theta| \leq \frac{\pi}{4}$.

Выбранные значения параметров модели квадрокоптера представлены в табл. 1.

Таблица 1

Параметры модели квадрокоптера

Параметр	Обозначение	Значение	Единица измерения
Масса	m	0.65	кг
Длина плеча	l	0.23	м
Момент инерции для оси *X*	J_x	7.5×10^{-3}	кг·м²
Момент инерции для оси *Y*	J_y	7.5×10^{-3}	кг·м²
Момент инерции для оси *Z*	J_z	1.3×10^{-2}	кг·м²
Коэффициент тяги	b	3.13×10^{-5}	Н/с²
Коэффициент крутящего момента	d	7.5×10^{-7}	Н·м/с²

Архитектура адаптивного управления

Модель «актор–критик»

Предложенная в [Ima22] архитектура представляет собой симбиоз самонастраивающегося ПИД-регулятора и глубокой нейронной сети типа «актор–критик», что позволяет сочетать преимущества обоих подходов. Общая структура системы управления показана на рис. 2.

Первый блок архитектуры – это модуль самонастраивающегося ПИД-регулятора. На этом этапе проектируется нейросеть, отвечающая за динамическую подстройку коэффициентов ПИД-регулятора. Затем управляющее воздействие вычисляется путем подачи ошибок (рассогласований) регулятора на вход этой сети. Полученное на каждом шаге управляющее воздействие вместе с последними выходными сигналами системы передается во второй блок архитектуры, который представляет собой сеть идентификации. Данная сеть, построенная по принципу «актор–критик», оценивает новое состояние системы. Фактически комплекс идентифи-

кации состоит из двух сетей. Первая – сеть Актора – предназначена для аппроксимации реального выхода системы. Вторая сеть, Критик, вычисляет функцию ценности входных данных сети (состояний среды) и показывает, насколько действие Актора целесообразно в текущем состоянии.

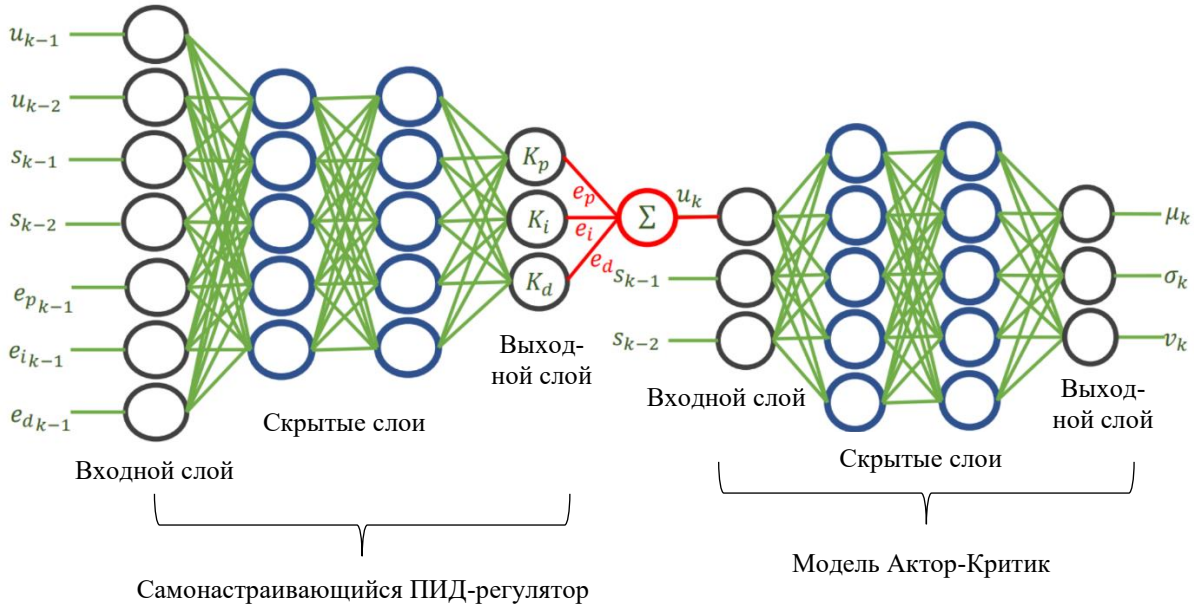

Рис. 2 Схематичное изображение нейронной сети [Ima22]

В скрытых слоях применяется сигмоидная функция активации, тогда как для слоя коэффициентов ПИД-регулятора используется гиперболический тангенс (tanh). В итоге динамические коэффициенты ПИД-регулятора (${K_n}^{\text{dynamic}}$) рассчитываются по следующей формуле:

$${K_n}^{\text{dynamic}}(k) = f_n(u(k-1), u(k-2), s(k-1), s(k-2), e_p(k-1), e_i(k-1), e_d(k-1)), \quad (5)$$

где $n = p, i, d$; $f_n(\cdot)$ – нелинейная функция с небольшим количеством весов и смещений, которые инициализируются вблизи нуля. В сети идентификации Актор имеет два выхода: математическое ожидание (μ) и дисперсию (σ). Эти выходные данные подаются в уравнение нормального распределения ($N(\mu\sigma^2)$), и из него случайным образом извлекается выборка (рис. 3).

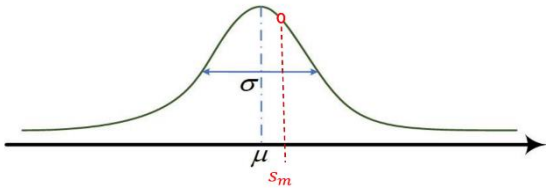

Рис. 3 Нормальное распределение [Ima22]

На рис. 3 s_m – это оценка каждого состояния квадрокоптера (углы ориентации и высота). Данная случайная величина является окончательным выходом сети Актора, и ожидается, что она будет соответствовать реальному выходному сигналу системы.

Задача Критика – оценить функцию ценности (v), используя состояния (управляющее воздействие и предыдущие выходы системы), тем самым предоставляя Актору обратную связь для улучшения его действий.

В конечном счете, выходные данные сети идентификации системы определяются следующими уравнениями:

$$\mu(k) = f_\mu(u(k), s(k-1), s(k-2)), \quad (6)$$

$$\sigma(k) = f_\sigma(u(k), s(k-1), s(k-2)), \quad (7)$$
$$v(k) = f_v(u(k), s(k-1), s(k-2)), \quad (8)$$

где $f_\mu(\cdot)$ и $f_\sigma(\cdot)$ – соответствующие функции Актора, а $f_v(\cdot)$ – функция Критика.

В результате объединения этих двух блоков выходной сигнал первого модуля подается на вход второго, формируя единую общую сеть (см. рис. 2). Входными данными для этой объединенной сети являются управляющие воздействия, текущие состояния и ошибки ПИД-регулятора. Выходами служат оценки каждого состояния и функция ценности входных параметров сети.

Модель Proximal Policy Optimization (PPO)

Данная архитектура также представляет собой симбиоз самонастраивающегося ПИД-регулятора и глубокой нейронной сети, однако вместо классического подхода «актор–критик» используется более современный алгоритм Proximal Policy Optimization (PPO) [Sch17], специально разработанный для повышения стабильности обучения в задачах глубокого обучения с подкреплением.

Общая структура системы управления сохраняется (см. рис. 2), где первый блок – модуль самонастраивающегося ПИД-регулятора – функционирует идентично описанному в модели «актор–критик». Динамические коэффициенты ПИД-регулятора рассчитываются по формуле (5), и полученное управляющее воздействие вместе с выходными сигналами системы передается во второй блок архитектуры.

Второй блок представляет собой усовершенствованную сеть идентификации, построенную по принципу PPO. Как и в классическом подходе «актор–критик», комплекс идентификации состоит из двух сетей: Актора (политики) и Критика (функции ценности). Однако PPO вводит ключевое улучшение – механизм ограничения изменения политики, предотвращающий дестабилизирующие большие шаги обновления.

Актор, аналогично предыдущей архитектуре, имеет два выхода: математическое ожидание (μ) и дисперсию (σ), которые подаются в уравнение нормального распределения ($N(\mu\sigma^2)$) для случайной выборки окончательного выхода (см. рис. 3). Критик оценивает функцию ценности состояния (v).

Ключевое отличие PPO заключается в специальной функции потерь для обновления Актора, которая включает ограничивающий член (clipping) [Sch17]:

$$L^{CLIP}(\theta) = E_t[\min(r_t(\theta)A_t, \text{clip}(r_t(\theta), 1-\epsilon, 1+\epsilon)A_t)],$$

где $r_t(\theta) = \frac{\pi_\theta(a_t|s_t)}{\pi_{\theta_{old}}(a_t|s_t)}$ – отношение вероятностей действий новой и старой политики; A_t – оценка преимущества (advantage); ϵ – гиперпараметр ограничения (выбран равным 0.2).

Выходные данные сети идентификации определяются уравнениями (6), (7) и (8), аналогично модели «актор–критик». Однако процесс обучения существенно отличается: PPO собирает траектории за несколько эпизодов, вычисляет преимущества, а затем выполняет несколько эпох оптимизации с ограниченными обновлениями политики.

В результате объединения блоков формируется единая сеть (см. рис. 2), где входными данными являются управляющие воздействия, текущие состояния и ошибки ПИД-регулятора, а выходами – оценки каждого состояния и функция ценности. Алгоритм PPO обеспечивает более стабильное обучение за счет предотвращения резких изменений политики, что особенно важно для таких чувствительных систем, как квадрокоптер.

В предложенной архитектуре используется схема распределенного обучения с независимыми агентами. Для каждого управляющего параметра: высоты (z) и углов ориентации (крена, тангажа и рыскания), функционирует выделенный нейросетевой агент, представляющий собой отдельную модель («актор–критик» или PPO). Данные агенты имеют идентичную структуру, но независимые параметры, и обучаются параллельно, специализируясь на управлении

исключительно своим целевым состоянием системы. Такой подход позволяет декомпозировать сложную задачу управления многомерной системой на набор более простых подзадач.

Оптимизация

После определения архитектуры нейронной сети и инициализации весовых коэффициентов и смещений требуется настроить ее параметры с помощью алгоритма оптимизации.

Основная задача Актора заключается в аппроксимации выходного сигнала системы, то есть в минимизации ошибки предсказания ($s_m - s$). В свою очередь, цель Критика – снизить ошибку временной разницы (Temporal Difference error, δ_{TD}) [Sut18]. Данная ошибка вычисляется по формуле (9), где γ представляет собой коэффициент дисконтирования, R_{k+1} – функцию вознаграждения, а v_k и v_{k+1} – значения функции ценности на шаге k и на следующем шаге соответственно.

$$\delta_{TD} = R_{k+1} + \gamma\, v_{k+1} - v_k. \tag{9}$$

Функция вознаграждения строится как квадратичная. Ее значение тем выше, чем меньше абсолютная ошибка, скорость ее изменения и величина управляющего сигнала, в соответствии с выражением (10). В этом уравнении r_1, r_2, r_3 – произвольные коэффициенты вознаграждения, а u – управляющее воздействие.

$$R_{k+1} = -r_1(s_m - s)^2 - r_2(\dot{s_m} - \dot{s})^2 - r_3 u^2\ . \tag{10}$$

Для обучения сети определяются две функции потерь: одна для Актора (L_a), а другая для Критика (L_c), как показано в уравнениях (11). Коэффициенты $\omega_1, \omega_2, \omega_3$ – это постоянные веса, задающие значимость каждого компонента функции. Параметр η, значение которого близко к нулю, введен для предотвращения обнуления потерь Актора в случае, когда δ_{TD} стремится к нулю. Это позволяет Актору продолжать исследование среды до достижения оптимальной стратегии.

$$L_a = \omega_1(s_m - s)^2(\eta + |\delta_{TD}|) + \omega_2\sqrt{2\pi e\sigma^2}, \quad L_c = \omega_3{\delta_{TD}}^2. \tag{11}$$

Критик, вычисляя сигнал δ_{TD}, передает его Актору, тем самым оценивая целесообразность выбранного действия. Высокое значение δ_{TD} увеличивает функцию потерь Актора, указывая на необходимость выбора иной стратегии. Если же δ_{TD} близка к нулю, это сигнализирует о достижении почти оптимального действия и способствует сходимости алгоритма.

Общая архитектура сети, объединяющая модуль самонастройки и идентификации, представлена на рис. 2. Для ее оптимизации используется совокупная функция потерь (L_t), получаемая суммированием потерь Актора и Критика:

$$L_t = L_a + L_c\ . \tag{12}$$

Модель использует оптимизатор Adam (Adaptive Moment Estimation) [Kin15]. Это распространенный [Zho19, Boc19] алгоритм оптимизации, который применяется для корректировки весов модели в процессе обучения нейронных сетей. Adam является одним из наиболее эффективных алгоритмов оптимизации в обучении нейронных сетей, который объединяет преимущества двух других оптимизаторов: адаптивного градиентного спуска (Adagrad) и стохастического градиентного спуска с инерцией (SGD with momentum). При этом данный подход сочетает в себе идеи RMSProp и оптимизатора импульса.

В отличие от RMSProp, который адаптирует скорость обучения параметров на основе среднего первого момента, Adam использует среднее значение вторых моментов градиентов. В частности, алгоритм вычисляет экспоненциальное скользящее среднее значение градиента и квадратичный градиент.

Следующие уравнения (13) описывают работу оптимизатора Adam:

$$m_t = \beta_1 m_{t-1} + (1-\beta_1)g_t,\ \ v_t = \beta_2 v_{t-1} + (1-\beta_2)g_t^2,$$

$$\widehat{m_t} = \frac{m_t}{1-\beta_1}, \quad \widehat{v_t} = \frac{v_t}{1-\beta_2}, \tag{13}$$
$$w_{t+1} = w_t - \frac{\alpha}{\sqrt{\widehat{v_t}+\varepsilon}}\widehat{m_t}, \quad g_t = \frac{\partial L_t}{\partial w_t}.$$

Определим, что w_{pid} и w_{ac} – веса самонастраивающегося ПИД-регулятора и системы «актор–критик» соответственно. Таким образом, скорость изменения функции суммарных потерь по каждому параметру вычисляется следующим образом:

$$g_{pid} = \left(\frac{\partial L_a}{\partial s_m}\frac{\partial s_m}{\partial u} + \frac{\partial L_a}{\partial \sigma}\frac{\partial \sigma}{\partial u} + \frac{\partial L_c}{\partial v}\frac{\partial v}{\partial u}\right)\frac{\partial u}{\partial w_{pid}}, \tag{14}$$

$$g_{ac} = \frac{\partial L_a}{\partial s_m}\frac{\partial s_m}{\partial w_{ac}} + \frac{\partial L_a}{\partial \sigma}\frac{\partial \sigma}{\partial w_{ac}} + \frac{\partial L_c}{\partial v}\frac{\partial v}{\partial w_{ac}}, \tag{15}$$

где

$$\frac{\partial u}{\partial w_{pid}} = e_p\frac{\partial K_p{}^{\text{dynamic}}}{\partial w_{\text{pid}}} + e_i\frac{\partial K_i{}^{\text{dynamic}}}{\partial w_{\text{pid}}}e_d\frac{\partial K_d{}^{\text{dynamic}}}{\partial w_{\text{pid}}}. \tag{16}$$

Важно упомянуть, что весовые коэффициенты e_p, e_i, e_d задаются извне и не являются объектом оптимизации.

Разработанная архитектура изначально предназначена для систем с одним входом и одним выходом (SISO). Однако квадрокоптер представляет собой многоканальную систему (MIMO). Данное противоречие разрешается путем декомпозиции: вблизи точки равновесия динамику аппарата можно разделить на четыре независимые SISO-подсистемы, отвечающие за углы ϕ, θ, ψ и высоту полета z. При этом подсистемы крена (ϕ) и тангажа (θ) являются достаточными для осуществления управления положением квадрокоптера по осям X и Y в связанной системе координат.

Результаты моделирования

Описанные в предыдущих разделах методы были реализованы на языке программирования Python с помощью библиотек PyTorch и других. Также была реализована симуляция с помощью CoppeliaSim для наглядности результатов. В качестве демонстрации работы методов были выбраны несколько траекторий для пролёта квадрокоптера: квадратная, фигура «Восьмёрка», «Слалом», трехмерная синусоида и траектория с резкими поворотами.

В качестве начального примера была задана траектория в форме квадрата на фиксированной высоте (рис. 4). В начале полета разброс (σ) при подборе коэффициентов велик, что свидетельствует об активном поиске агентом оптимальных действий для улучшения идентификации модели. Это приводит к значительным колебаниям значений коэффициентов ПИД-регулятора, которые со временем затухают, стабилизируясь в окрестности постоянных величин. На рис. 5 и 7 представлены графики углов крена и тангажа для обоих методов, и можно видеть, что они справляются с задачей одинаково стабильно. Графики подтверждают, что предложенный в [Ima22] алгоритм успешно выполняет задачу стабилизации ориентации квадрокоптера в рамках данного сценария.

Мониторинг функции вознаграждения и потерь (рис. 6) демонстрирует их стабилизацию с течением времени до достижения оптимума в случае метода «актор–критик». Эта динамика указывает на успешную оптимизацию весов сети, которая проводилась в режиме онлайн. Такой подход обеспечивает высокое быстродействие метода, что позволяет применять его на реальных роботизированных платформах.

В то время как метод «актор–критик» демонстрировал классическую монотонную сходимость, функция потерь PPO вела себя несколько иным образом. Относительно оси Z функция потерь демонстрирует в некотором роде случайное поведение, постепенно снижаясь, а относительно углов – быстро стабилизируется.

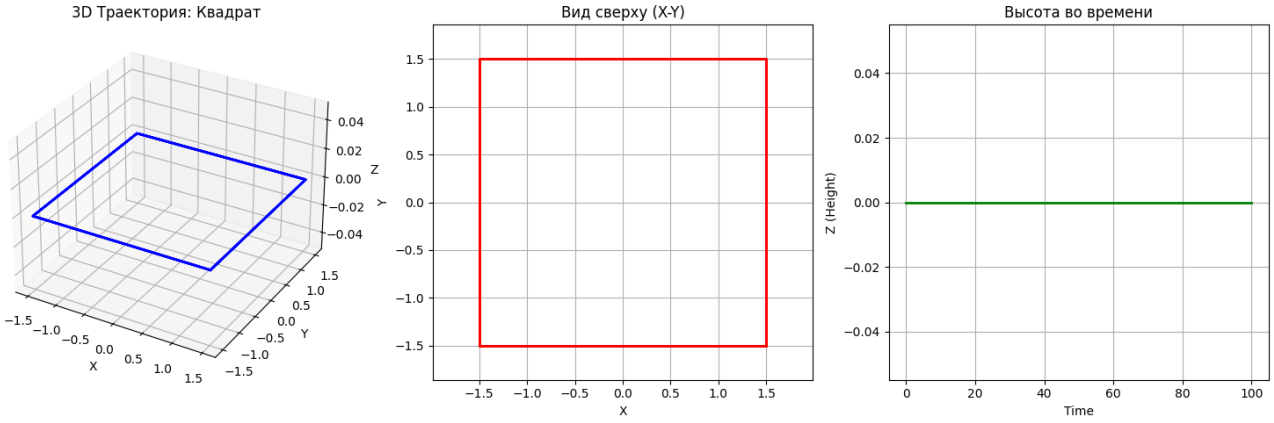

Рис. 4 Визуализация траектории «Квадрат»

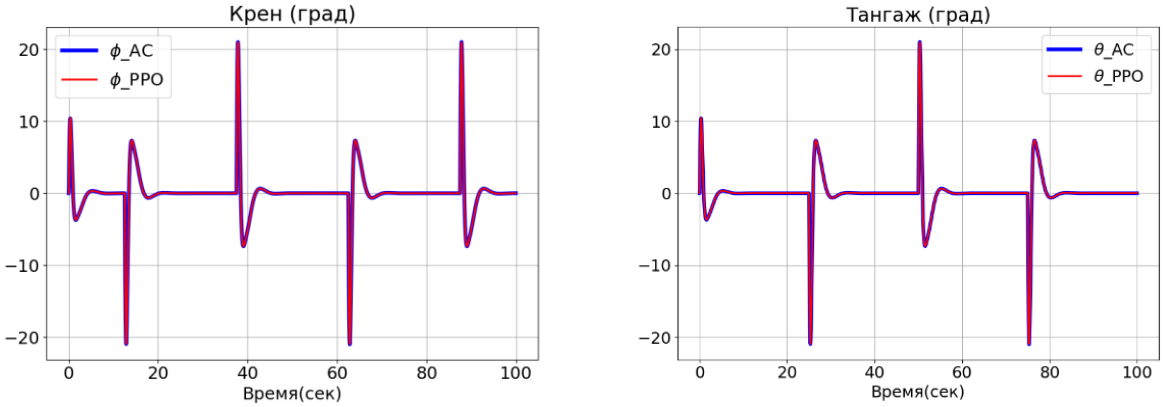

Рис. 5 Контроль углов крена и тангажа
в случае квадратной траектории

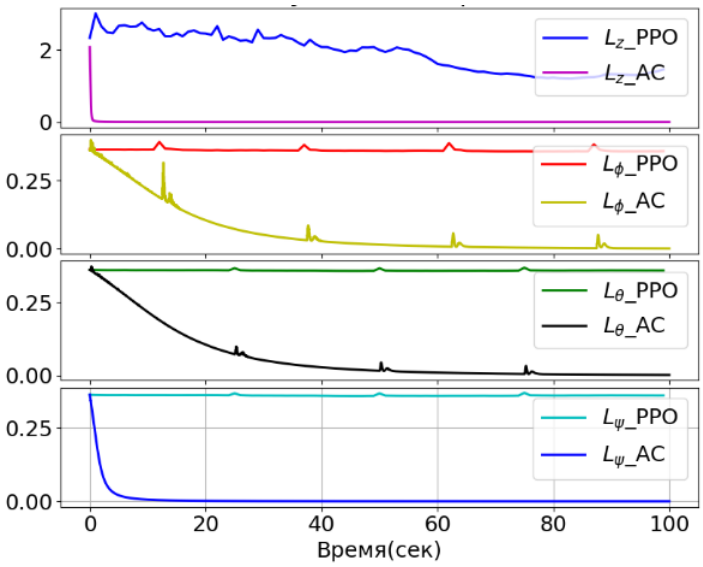

Рис. 6 Изменения функций потерь
для сети агентов во времени для квадратной траектории

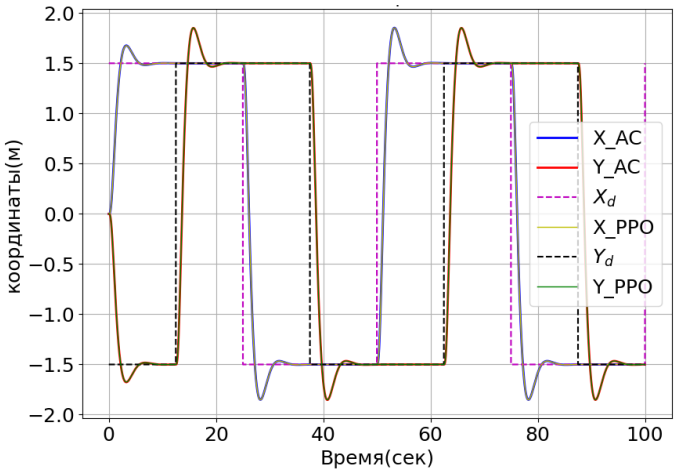

Рис. 7 Изменения координат квадрокоптера по осям X и Y во времени при движении по квадратной траектории

Данное явление объясняется фундаментальным отличием в природе оптимизируемых функций. В методе «актор–критик» функция потерь напрямую отражает ошибку политики и её минимизацию. В PPO оптимизируется суррогатная цель, которая является аппроксимацией ожидаемого улучшения политики с ограничениями. Найденная PPO политика оказалась близка к локальному оптимуму, в окрестностях которого алгоритм продолжает поиск, но ограничивающий механизм (clipping) не позволяет сделать слишком большие шаги, что и проявляется в колебаниях суррогатной функции для оси Z и её стабилизации для углов, где стратегия уже нашла устойчивое решение.

Наблюдаемое расхождение между динамикой функции потерь и фактической производительностью политики согласуется с выводами других исследований, применяющих алгоритм PPO. Как отмечают создатели алгоритма [Sch17], суррогатная цель L^{CLIP} является зашумленным прокси для истинного показателя эффективности, и её значение может ухудшаться даже при улучшении политики. Данное поведение дополнительно подчеркивается в современных эмпирических исследованиях [Ber19, Eng20], где показано, что успешное обучение с помощью PPO сильно зависит от деталей реализации и корректной интерпретации метрик, при этом мониторинг среднего вознаграждения за эпизод является более надежным индикатором, чем значение функции потерь.

Полученные результаты отмечают, что в обучении с подкреплением, в отличие от классического контролируемого обучения, динамика функции потерь не всегда является прямым индикатором качества обученной политики. PPO может демонстрировать «нестабильные» графики потерь, при этом успешно решая поставленную задачу.

После настройки ПИД-коэффициентов в первом сценарии система управления углами Эйлера начинает точно отслеживать заданные значения. Достижение адекватного управления ориентацией позволило легко реализовать и точное позиционирование аппарата в пространстве, что иллюстрирует рис. 7.

Для оценки эффективности алгоритмов при выполнении сложных плавных манёвров была выбрана траектория в виде фигуры «Восьмёрка» (рис. 8). Как видно из графиков на рис. 9, система уверенно выполняет стабилизацию углов крена и тангажа, даже несмотря на непрерывные изменения курса. Эволюция функции вознаграждения и потерь (рис. 10) подтверждает корректную работу алгоритма «актор–критик»: после начального периода адаптации наблюдается их устойчивая сходимость к оптимальным значениям, что свидетельствует об успешной онлайн-настройке весовых коэффициентов нейронной сети. Функции потерь же для метода PPO ведут себя так же нестандартно, как и ранее. Настроенные в первом эксперименте

ПИД-коэффициенты показали свою универсальность, обеспечив не только требуемую ориентацию, но и высокую точность следования по пространственной траектории, что наглядно демонстрирует рис. 11.

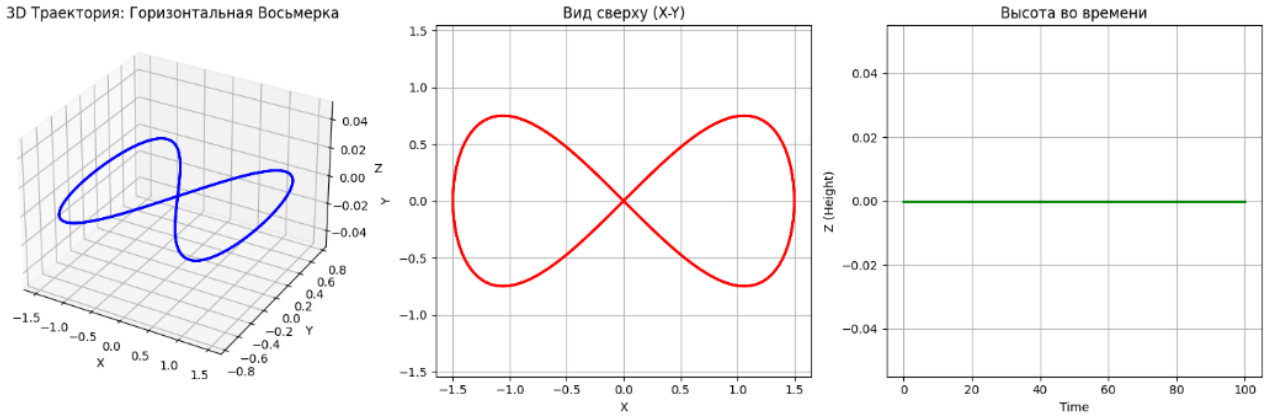

Рис. 8 Визуализация траектории «Восьмёрка»

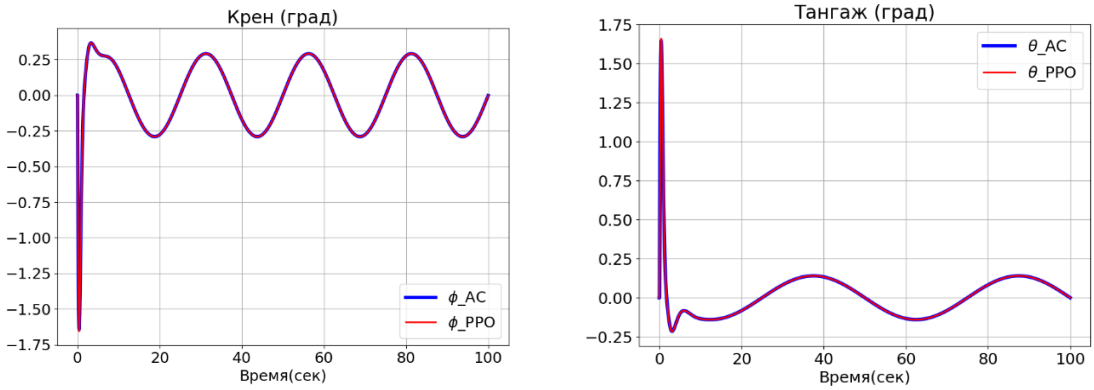

Рис. 9 Контроль углов крена и тангажа в случае траектории «Восьмёрка»

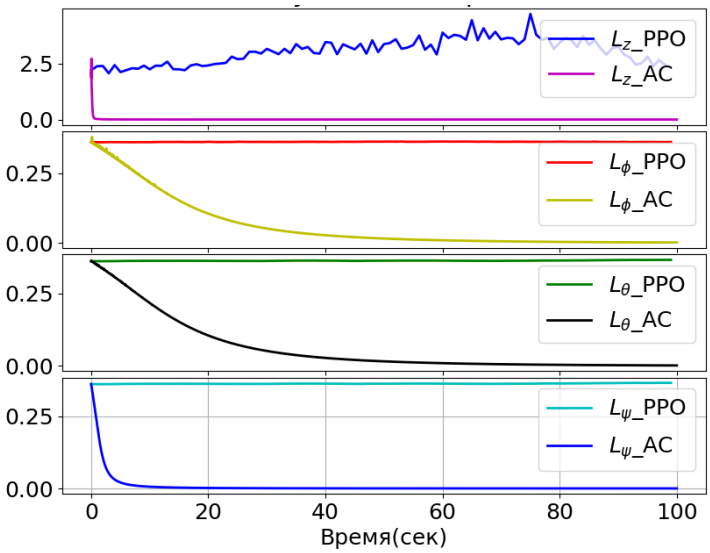

Рис. 10 Изменения функций потерь для сети агентов во времени для траектории «Восьмёрка»

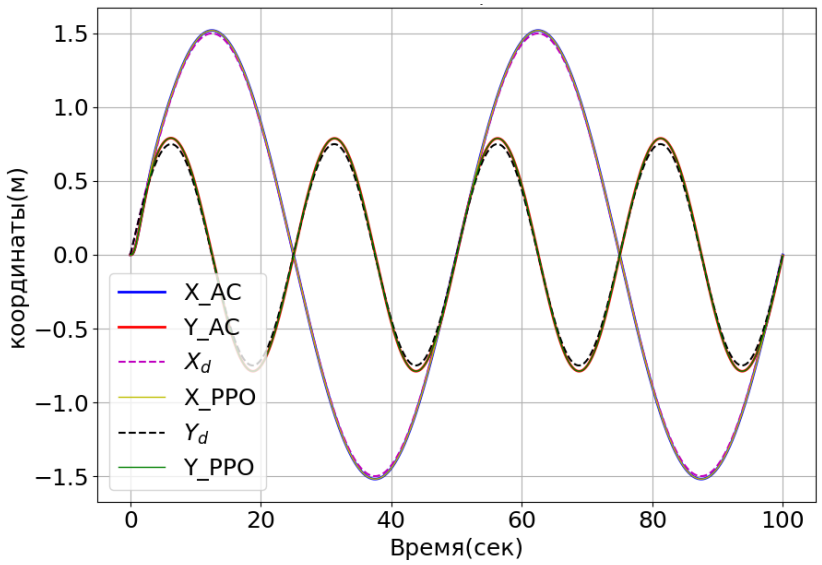

Рис. 11 Изменения координат квадрокоптера по осям *X* и *Y* во времени при движении по траектории «Восьмёрка»

Следующим этапом стала проверка алгоритмов в условиях, имитирующих объезд непредвиденных препятствий, – траектория «Слалом», заданная в виде сложной синусоиды с различными частотами по осям (рис. 12). Анализ графиков ориентации (рис. 13) показывает, что алгоритмы оперативно и точно реагируют на частые изменения полетного задания, обеспечивая стабилизацию летательного аппарата. Кривые обучения для метода «актор–критик» на рис. 14, несмотря на повышенную сложность маршрута, вновь демонстрируют характерную динамику: после этапа поиска происходит стабилизация вознаграждения и потерь, что указывает на непрерывную и эффективную оптимизацию нейронной сети в реальном времени. Функция потерь PPO для вертикальной оси *Z* проявляла осцилляционный характер, а для углов Эйлера – стабилизировалась. Однако в силу суррогатности данного параметра в данном случае на результат это не повлияло. Как и в предыдущих случаях, было достигнуто точное позиционирование в пространстве (рис. 15), что критически важно для автономной навигации в сложной обстановке.

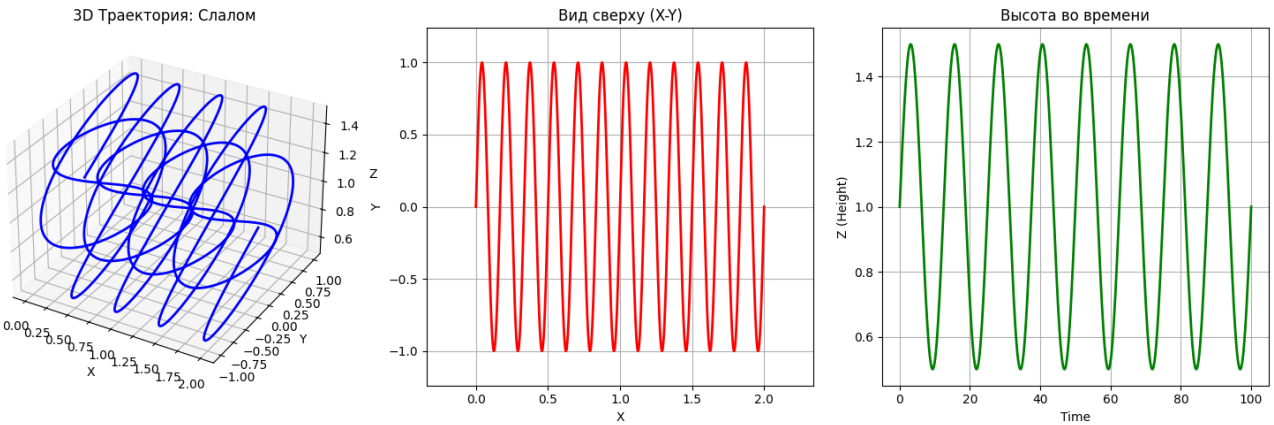

Рис. 12 Визуализация траектории «Слалом»

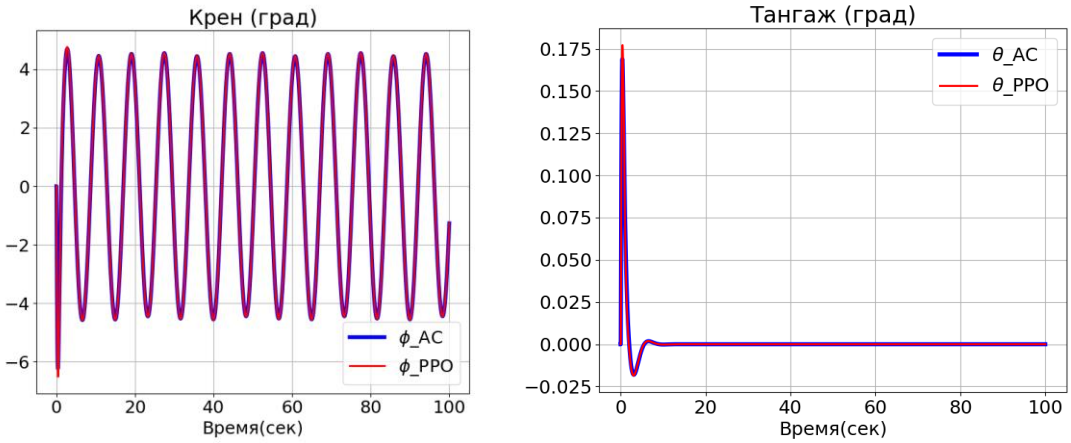

Рис. 13 Контроль углов крена и тангажа в случае траектории «Слалом»

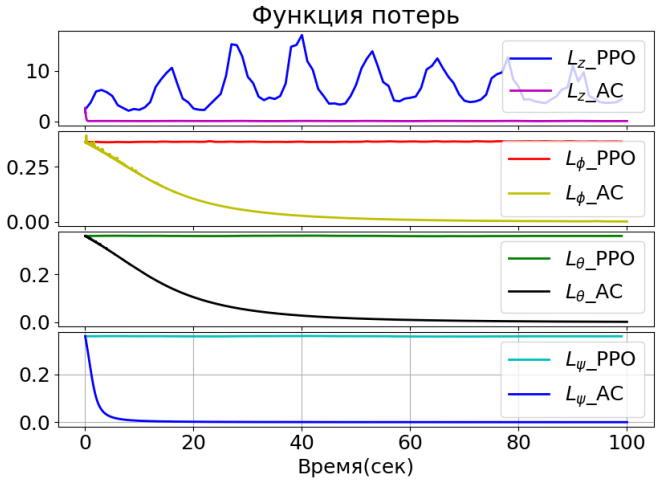

Рис. 14 Изменения функций потерь для сети агентов во времени для траектории «Слалом»

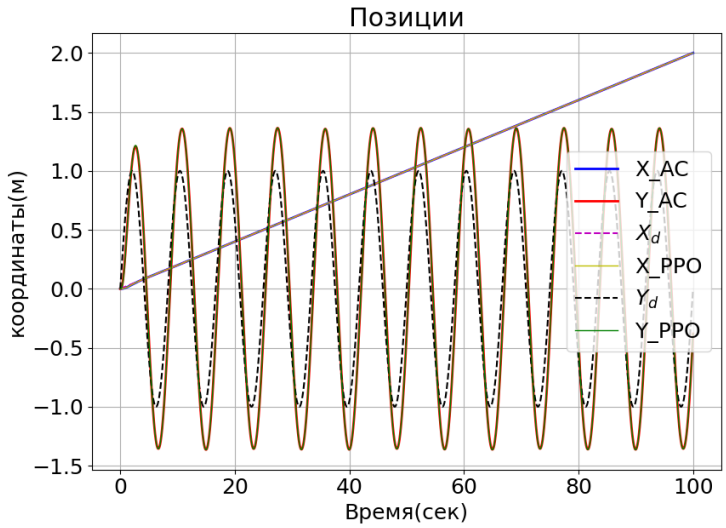

Рис. 15 Изменения координат квадрокоптера по осям X и Y во времени при движении по траектории «Слалом»

Для комплексной проверки системы управления была использована трёхмерная синусоидальная траектория, предполагающая одновременное и согласованное изменение положения по всем осям (рис. 16). Данный сценарий проверяет способность алгоритма координировать многосвязные движения. Результаты, представленные на рис. 17, подтверждают, что квадрокоптер сохраняет устойчивость и управляемость даже при таком сложном характере полёта. Мониторинг показателей обучения метода «актор–критик» (рис. 18) фиксирует плавный выход на оптимум, подчёркивая способность алгоритма адаптироваться к многосвязным задачам. Следствием успешного управления ориентацией стала и точная отработка пространственного пути, что иллюстрирует рис. 19.

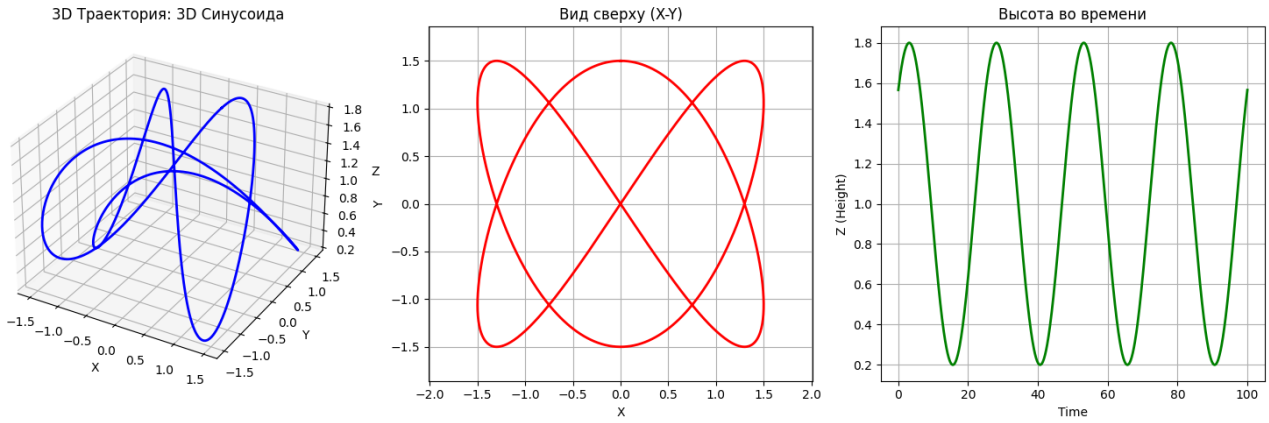

Рис. 16 Визуализация траектории трёхмерной синусоиды

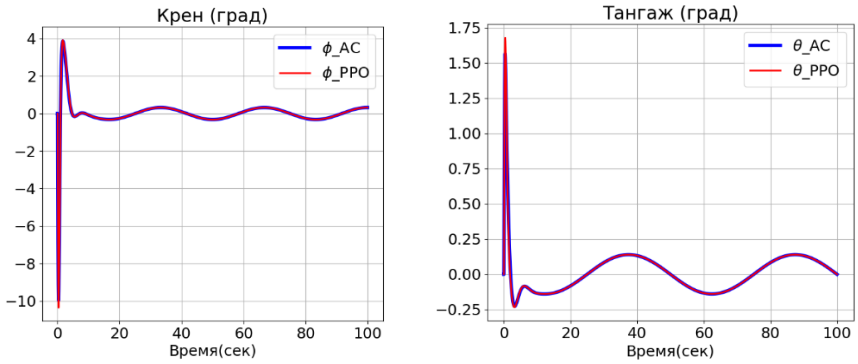

Рис. 17 Контроль углов крена и тангажа в случае траектории трёхмерной синусоиды

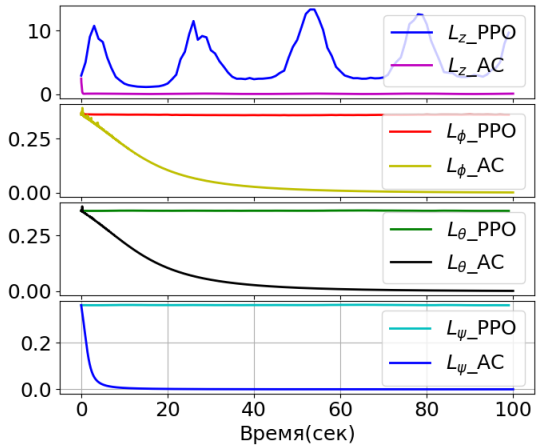

Рис. 18 Изменения функций потерь для сети агентов во времени для траектории трёхмерной синусоиды

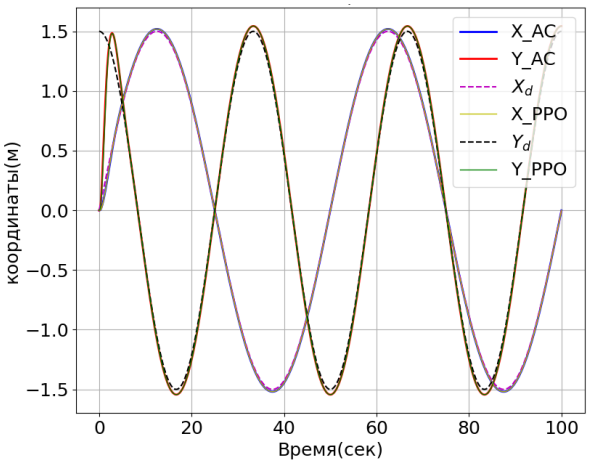

Рис. 19 Изменения координат квадрокоптера по осям X и Y во времени при движении по траектории трёхмерной синусоиды

Финальный тест в отсутствие шумов моделировал экстремальную ситуацию – траекторию с резкими поворотами на 150 градусов (рис. 20). Целью была проверка запаса устойчивости и быстродействия системы. Полученные данные (рис. 21) наглядно демонстрируют, что алгоритм эффективно парирует столь значительные рассогласования, успешно стабилизируя ориентацию. Динамика функции вознаграждения и потерь для метода «актор–критик» (рис. 22), несмотря на агрессивный характер манёвров, показывает устойчивую тенденцию к оптимизации, что свидетельствует о корректной работе механизма онлайн-обучения. В результате система обеспечила точное отслеживание заданной траектории (рис. 23), подтвердив свою надёжность.

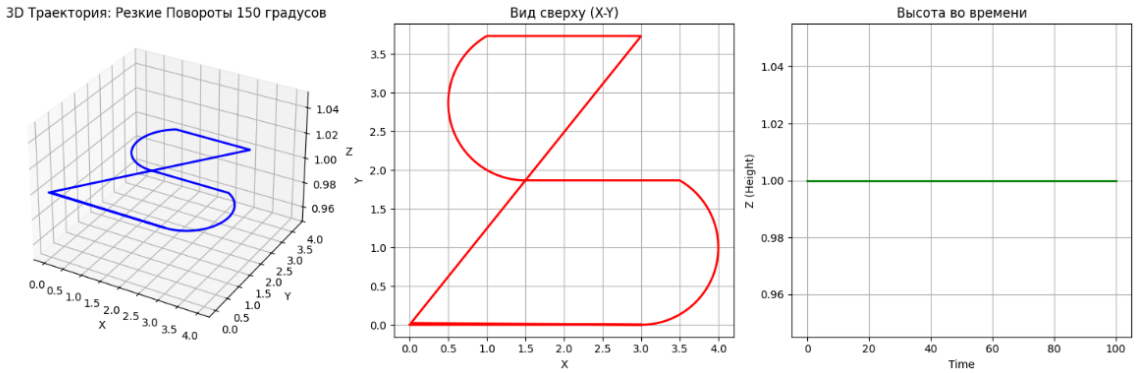

Рис. 20 Визуализация траектории с резкими поворотами

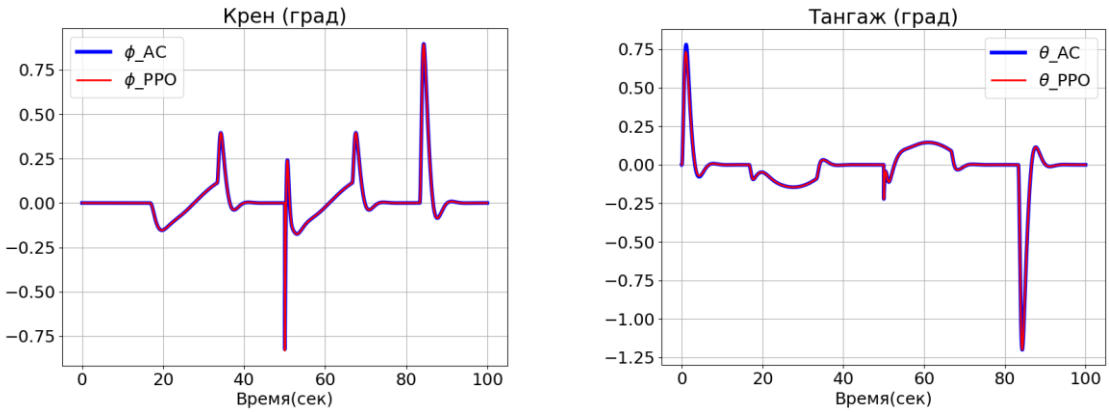

Рис. 21 Контроль углов крена и тангажа в случае траектории с резкими поворотами

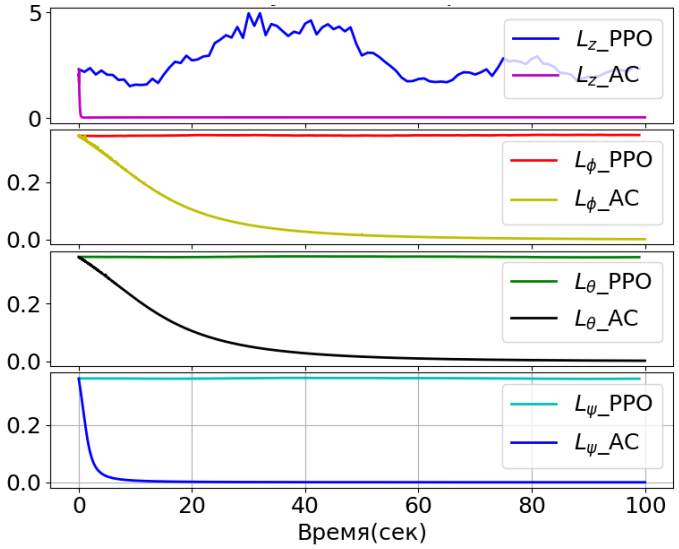

Рис. 22 Изменения функций потерь для сети агентов во времени для траектории с резкими поворотами

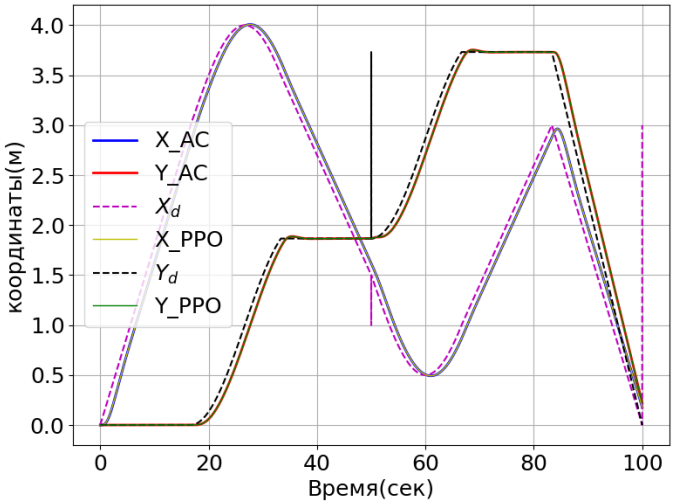

Рис. 23 Изменения координат квадрокоптера по осям X и Y во времени при движении по траектории с резкими поворотами

Для проверки устойчивости системы к изменениям параметров в процессе движения и оценки эффективности представленного алгоритма была задана динамическая вариация полной массы квадрокоптера (рис. 24). Моделирование заключается в резком изменении массы аппарата через короткий промежуток времени. Рост массы приводит к изменению высоты, что требует от системы управления способности адаптироваться к новым условиям. Как видно из рис. 24, система успешно компенсирует возникающую ошибку (рассогласование). Небольшие колебания высоты в процессе полета объясняются тем, что моделирование данного случая происходило при пролёте дрона по квадратной траектории.

Традиционный ПИД-регулятор, параметры которого настроены для системы с фиксированной массой, не может справиться с возмущениями такого рода [Pou12]. В то же время разработанный алгоритм оперативно корректирует свои коэффициенты, что не позволяет ошибке накапливаться. Это подтверждает, что метод обладает свойствами адаптивности и способности к онлайн-самонастройке.

Также стоит отметить, что метод «актор–критик» справился с изменениями массы лучше, чем метод PPO. Хотя к изначальной высоте дрон возвращался почти одновременно в обоих методах, отклонение от этой высоты было значительно сильнее для метода PPO.

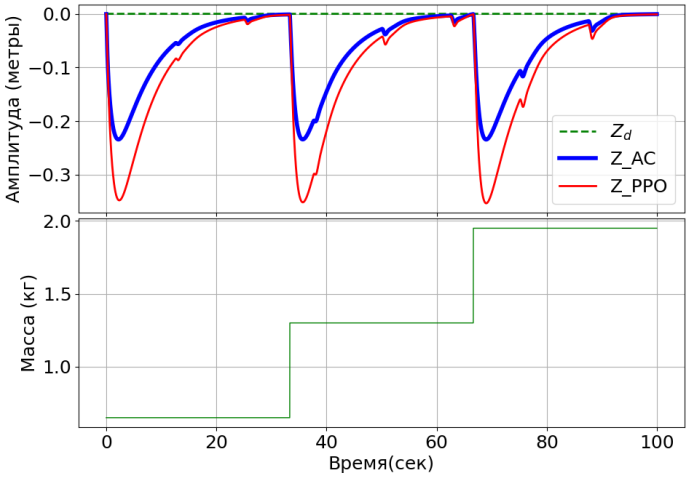

Рис. 24 Стабилизация высоты полета квадрокоптера при изменении массы

Также для проверки устойчивости системы ко внешним возмущениям в процессе движения и оценки эффективности алгоритма в модель были добавлены возмущения Гаусса–Маркова [Din21] с помощью следующего уравнения:

$$\dot{d} = -\frac{1}{\tau_s} d + \rho B_w q_w. \tag{17}$$

Эта модель имитирует ветер, меняющийся порывами. Уравнение (17) известно как «формирующий фильтр» для порывов ветра, где q_w – независимая постоянная с нулевым средним значением; $\tau_s = 0.3$ – время корреляции ветра» B_w – входная идентифицирующая матрица турбулентности; $\rho = 0.5$ – скалярный весовой коэффициент. На рис. 25 показаны зарегистрированные возмущения Гаусса–Маркова во время полета квадрокоптера при нулевых начальных условиях. Величина возмущений достаточна, чтобы повлиять на эффективность стабилизации движения.

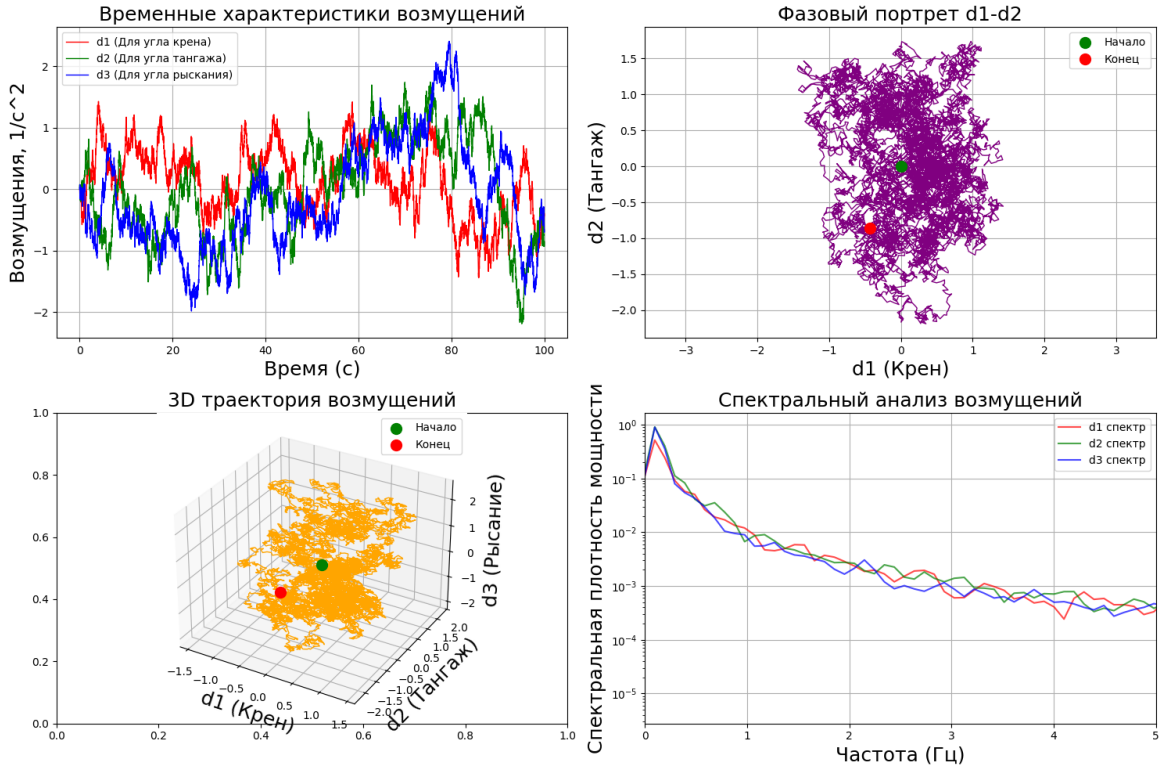

Рис. 25 Возмущения Гаусса–Маркова,
значения $d_1,\ d_2,\ d_3$ относятся соответственно к углам ϕ, θ, ψ

В результате проверки работы методов в таких условиях они показали также практически идентичные результаты, заключающиеся в некоторых отклонениях от желаемой траектории (рис. 26–30).

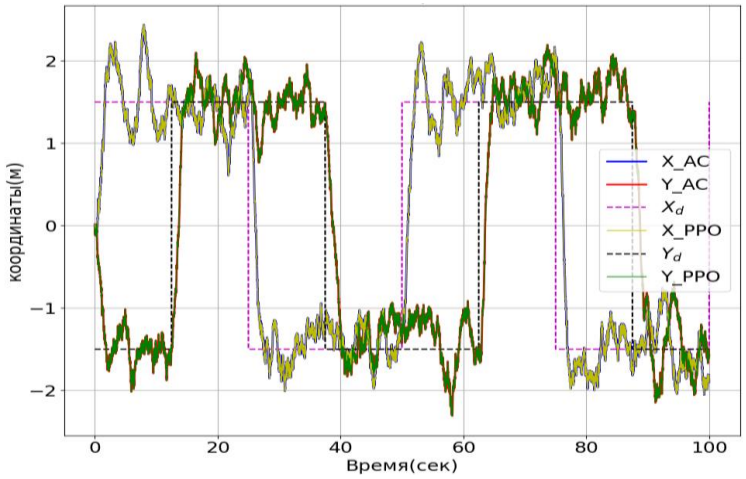

Рис. 26 Изменения координат квадрокоптера по осям X и Y во времени при движении по квадратной траектории при влиянии возмущений Гаусса–Маркова

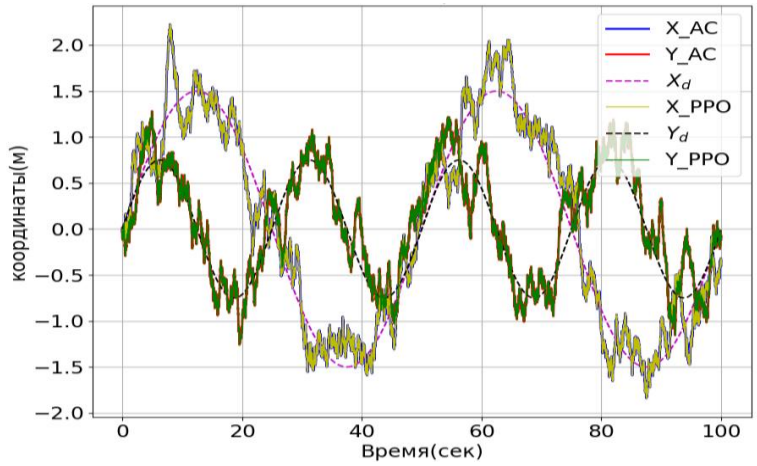

Рис. 27 Изменения координат квадрокоптера по осям X и Y во времени при движении по траектории «Восьмёрка» при влиянии возмущений Гаусса–Маркова

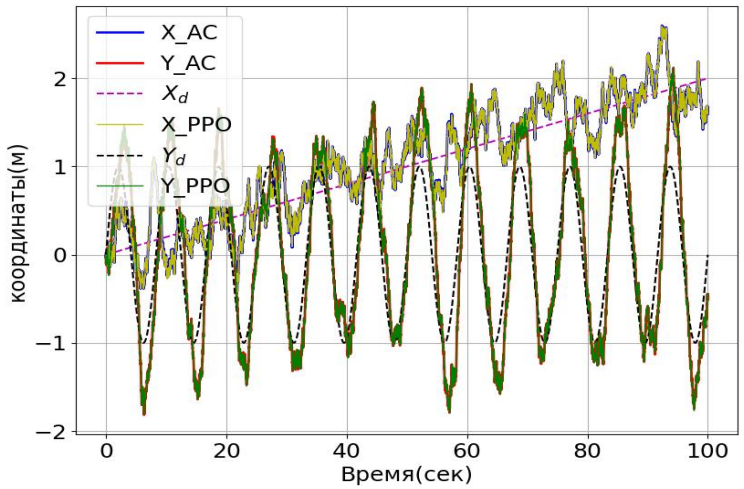

Рис. 28 Изменения координат квадрокоптера по осям X и Y во времени при движении по траектории «Слалом» при влиянии возмущений Гаусса–Маркова

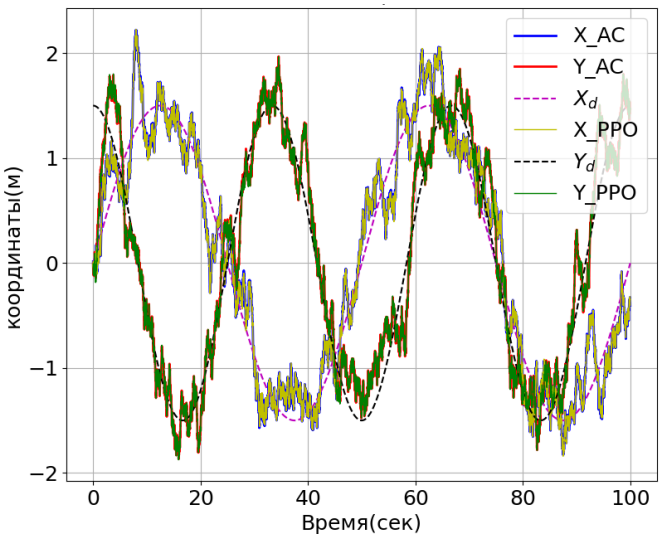

Рис. 29 Изменения координат квадрокоптера по осям X и Y во времени при движении по траектории трёхмерной синусоиды при влиянии возмущений Гаусса–Маркова

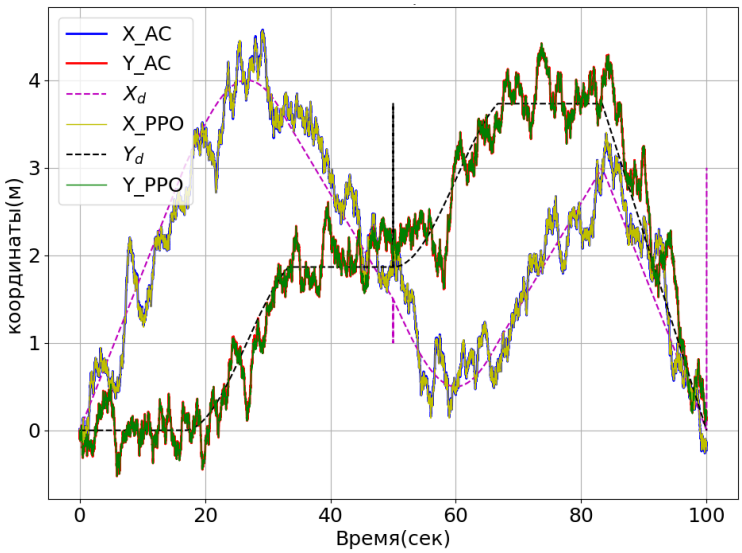

Рис. 30 Изменения координат квадрокоптера по осям X и Y во времени при движении по траектории с резкими поворотами при влиянии возмущений Гаусса–Маркова

По сравнению с ПИД-регулятором, жестко заданные коэффициенты которого со временем приводят к значительной нескомпенсированной ошибке, данные методы адаптивно перестраивают свои параметры. Это достигается за счет того, что регулятор в реальном времени оптимизирует коэффициенты, опираясь на динамику ошибки, её скорость, величину управляющего сигнала и историю состояний системы.

Для более наглядного сравнения предложенных методов были посчитаны RMSE-ошибки для каждой траектории при влиянии возмущений Гаусса–Маркова. Результаты приведены в табл. 2.

Как видно из табл. 2, отличия проявляются практически всегда только после 4 знака после запятой, что несущественно. Однако средняя разница во времени обучения архитектур соста-

вила 83.33, так как среднее время обучения для «актор–критик» метода составило 131.21 секунды и 47.88 секунды для PPO метода. Таким образом, время обучения у метода, основанного на PPO, оказалось примерно в 2.8 раза меньше, чем у метода, основанного на подходе «актор–критик». Поэтому можно сделать вывод, что эффективность работы метода PPO в части времени обучения оказалась выше, чем метода «актор–критик», при тех же результатах.

Таблица 2

Ошибки RMSE для двух методов

Траектория	A2C	PPO
Квадрат	0.6068	0.6071
Фигура «Восьмёрка»	0.2786	0.2787
«Слалом»	0.2799	0.2800
Трёхмерная синусоида	0.3183	0.3186
Резкие повороты	0.2784	0.2787

Заключение

В данной работе представлено сравнение метода [Ima22] самонастройки ПИД-регулятора, основанного на гибридной нейросетевой архитектуре по схеме «актор–критик», с методом такой же самонастройки, но основанной на методе Proximal Policy Optimization (PPO) [Sch17]. Разработанные подходы позволяют в реальном времени использовать адаптированные коэффициенты регулятора и идентифицировать состояния системы. Методы не только отличаются относительно простой структурой, но и применимы к реальным объектам управления с одним входом и выходом (SISO). В исследовании использованы преимущества нейронных сетей и оптимизатора Adam, обеспечивающего высокую скорость и надежность вычислений.

Результаты моделирования продемонстрировали способность алгоритмов отслеживать сложные траектории даже при случайной начальной инициализации весов. Методы показали эффективность управления параметрами квадрокоптера на траекториях разной сложности, не выявив значительного отклонения от идеального маршрута.

Также система управления продемонстрировала устойчивость к изменениям параметров как внутренних, так и внешних – изменение массы и порывы ветра соответственно. При этом метод, использующий архитектуру «актор–критик», справился с возмущениями массы эффективнее, чем метод, использующий архитектуру PPO.

Результаты прямого сравнения методов не выявили значительной разницы между ними, за исключением влияния изменения массы на высоту – ошибка RMSE даже в условиях возмущений Гаусса–Маркова не показала существенных отличий. Однако время обучения архитектуры PPO оказалось почти в 3 раза меньше, чем аналогичное время для архитектуры «актор–критик».

Таким образом, методы могут быть использованы в дальнейшей разработке системы управления квадрокоптером. При этом метод PPO является предпочтительным в условиях сложных систем, долгих полетов и тому подобного в силу эффективности обучения, а метод «актор–критик» является предпочтительным в условиях изменения массы.

В дальнейшем планируется провести сравнение метода, основанного на ПИД+PPO системе, с другими подходами, использующими, например, метод Soft Actor Critic (SAC), который также доказал свою эффективность при работе с квадрокоптерами [Mah24].

Благодарности и поддержка

Работа выполнена в рамках Госзадания FMRS-2023-0016 (123020700078-8).

СПИСОК ЛИТЕРАТУРЫ | REFERENCES

[Åst22] Åström K. J., Hägglund T. The future of PID control // Control Engineering Practice. 2001. Vol. 9, Is. 11. Pp. 1163–1175. URL: https://doi.org/10.1016/S0967-0661(01)00062-4

[Ban16] Bannwarth J. X. J., Chen Z. J., Stol K. A., MacDonald B. A. Disturbance accomodation control for wind rejection of a quadcopter // International Conference on Unmanned Aircraft Systems (ICUAS), Arlington, VA, USA. 2016. URL: https://doi.org/10.1109/ICUAS.2016.7502632

[Ber19] Berner C., Brockman G. et. al. Dota 2 with Large Scale Deep Reinforcement Learning // OpenAI. 2019. URL: https://doi.org/10.48550/arXiv.1912.06680

[Boc19] Bock S., Weiß M. A proof of local convergence for the Adam optimizer // International Joint Conference on Neural Networks (IJCNN), IEEE. 2019 https://doi.org/10.1109/IJCNN.2019.8852239

[Din21] Ding L., He Q., Wang C., Qi R. Disturbance rejection attitude control for a quadrotor: Theory and experiment // International Journal of Aerospace Engineering. 2021. Vol. 2. Pp. 1–15. URL: https://doi.org/10.1155/2021/8850071.

[Eng20] Engstrom L., Ilyas A., Santurkar S., Tsipras D., Janoos F., Rudolph L., Madry A. Implementation Matters in Deep Policy Gradients: A Case Study on PPO and TRPO // International Conference on Learning Representations (ICLR). 2020. URL: https://doi.org/10.48550/arXiv.2005.12729

[Fan19] Fan J., Saadeghvaziri M. Applications of Drones in Infrastructures: Challenges and Opportunities // International Journal of Mechanical, Industrial and Aerospace Sciences. 2019. Vol. 12, n. 10. URL: https://doi.org/10.5281/zenodo.3566281

[Gro12] Grondman I., Busoniu L., Lopes G. A. D., Babuska R. A Survey of Actor-Critic Reinforcement Learning: Standard and Natural Policy Gradients // IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews). 2012. Vol. 42, no. 6. Pp. 1291–1307. URL: https://doi.org/10.1109/TSMCC.2012.2218595

[Gup25] Gupta O. Precision Agriculture with Drones: A New Age of Farming // AkiNik Publications. 2025. 82 pp. URL: https://doi.org/10.22271/ed.book.3126

[Hen22] Henderson P., Islam R., Bachman P., Pineau J., Precup D., Meger D. Deep reinforcement learning that matters // Thirthy-Second AAAI Conference On Artificial Intelligence. 2018. URL: https://doi.org/10.1609/aaai.v32i1.11694

[Ima22] Iman S., Aria A. Self-Tuning PID Control via a Hybrid Actor-Critic-Based Neural Structure for Quadcopter Control // The 30th Annual International Conference of Iranian Society of Mechanical Engineers. Iran. 2022. URL: https://doi.org/10.48550/arXiv.2307.01312

[Kin15] Kingma D. P., Ba J. Adam: A method for stochastic optimization // 3rd International Conference for Learning Representations, San Diego. 2015. URL: https://doi.org/10.48550/arXiv.1412.6980

[LiY24] Li Y., Zhu Q., Elahi A. Quadcopter trajectory tracking control based on flatness model predictive control and neural network // Actuators. 2024. Vol. 13, 154. 20 p. URL: https://doi.org/10.3390/act13040154.

[Lop23] Lopez-Sanchez I., Moreno-Valenzuela J. PID control of quadrotor UAVs: A survey // Annual Reviews in Control. 2023. Vol. 56. P. 100900. URL: https://doi.org/10.1016/j.arcontrol.2023.100900.

[Mah24] Mahran Y., Gamal Z., El-Badawy A. Reinforcement Learning Position Control of a Quadrotor Using Soft Actor-Critic (SAC) // 6th Novel Intelligent and Leading Emerging Sciences Conference (NILES). IEEE. 2024. URL: https://doi.org/10.1109/NILES63360.2024.10753187

[Ngu21] Nguyen N. P., Mung N. X., Thanh H. L. N. N., Huynh T. T., Lam N. T., Hong S. K. Adaptive Sliding Mode Control for Attitude and Altitude System of a Quadcopter UAV via Neural Network // IEEE Access. 2021. Vol. 9. Pp. 40076–40085, URL: https://doi.org/10.1109/ACCESS.2021.3064883.

[Pou12] Pounds P. E. I., Bersak D. R., Dollar A. M. Stability of small-scale UAV helicopters and quadrotors with added payload mass under PID control // Auton Robot. 2012. 33. Pp. 129–142. URL: https://doi.org/10.1007/s10514-012-9280-5.

[Rum86] Rumelhart D. E., Hinton G. E., Williams R. J. Learning representations by back-propagating errors // Nature. 1986. Vol. 323 Pp. 533–536. URL: https://doi.org/10.1038/323533a0.

[Sch17] Schulman J., Wolski F., Dhariwal P., Radford A., Klimov O. Proximal Policy Optimization Algorithms // OpenAI. 2017. URL: https://doi.org/10.48550/arXiv.1707.06347.

[Sha20] Shahmoradi J., Talebi E., Roghanchi P., Hassanalian M. A Comprehensive Review of Applications of Drone Technology in the Mining Industry // Drones. 2020. Vol. 4, no. 3: 34. URL: https://doi.org/10.3390/drones4030034.

[Sut18] Sutton R. S., Barto A. G. Reinforcement learning: An introduction. // IEEE Transactions on Neural Networks. 1998. Vol. 9, Is 5. 1054 pp. URL: https://doi.org/10.1109/TNN.1998.712192.

[Tan18] Tangkaratt V., Abdolmaleki A., Sugiyama M. Guide Actor-Critic for Continuous Control // International Conference on Learning Representations (ICLR). 2018. URL: https://doi.org/10.48550/arXiv.1705.07606.

[Tri15] Tripathi V. K., Behera L., Verma N. Design of sliding mode and backstepping controllers for a quadcopter // 39th National Systems Conference (NSC). IEEE. 2015. URL: https://doi.org/10.1109/NATSYS.2015.7489097.

[Wah10] Waharte S., Trigoni N. Supporting Search and Rescue Operations with UAVs // International Conference on Emerging Security Technologies, Canterbury, UK. 2010. URL: https://doi.org/10.1109/EST.2010.31.

[Zha24] Zhang J., Rivera C. E. O., Tyni K., Nguyen S., Leal U. S. C., Shoukry Y. AirPilot Drone Controller: Enabling Interpretable On-the-Fly PID Auto-Tuning via DRL // IEEE 6th International Conference on Civil Aviation Safety and Information Technology (ICCASIT). 2024. URL: https://doi.org/10.1109/ICCASIT62299.2024.10828099.

[Zho19] Zhou J., Wang H., Wei J., Liu L., Huang X., Gao S., Liu W., Li J., Yu C., Li Z. Adaptive moment estimation for polynomial nonlinear equalizer in PAM8-based optical interconnects // Optics express. 2019. Vol. 27, No. 22. Pp. 32210–32216. URL: https://doi.org/10.1364/OE.27.032210

[Гур24] Гурчинский М. М., Тебуева Ф. Б. Обнаружение нарушителя агентами роевых робототехнических систем в условиях недетерминированной среды функционирования // СИИТ. 2024. Т. 6, № 3(18). С. 71–82. URL: https://doi.org/10.54708/2658-5014-SIIT-2024-no3-p71. EDN AUVYOX. [[Gurchinsky M. M., Tebueva F. B. Intruder Detection by Agents of Swarm Robotic Systems in a Non-Deterministic Operating Environment // SIIT. 2024. Vol. 6, No. 3(18). Pp. 71-82. (In Russian).]]

[Мус24] Муслимов Т. З. Методы и алгоритмы группового управления беспилотными летательными аппаратами самолетного типа // СИИТ. 2024. Т. 6, № 1(16). С. 3–15. URL: https://doi.org/10.54708/2658-5014-SIIT-2024-no1-p3. EDN HOTUZU. [[Muslimov T. Z. Methods and Algorithms for Formation Control of Fixed-Wing Unmanned Aerial Vehicles // SIIT. 2024. Vol. 6, No. 1(16). Pp. 3-15. (In Russian).]]

[При25] Приходько В. Е., Тепляшин П. Н., Плотников А. В., Шебухова О. А. Практическая реализация коммуникационной системы мобильной группы на основе нейронных сетей // СИИТ. 2025. Т. 7, № 1(20). С. 96–104. URL: https://doi.org/10.54708/2658-5014-SIIT-2025-no1-p96. EDN UYDDVC. [[Prikhodko V. E., Teplyashin P. N., Plotnikov A. V., Shebukhov O. A. Practical Implementation of a Mobile Group Communication System Based on Neural Networks // SIIT. 2025. Vol. 7, No. 1(20). Pp. 96-104. (In Russian).]]

[Саи25] Саитова Г. А., Габдуллина Э. Р. Методика определения проективного покрытия полей на основе дистанционного мониторинга // СИИТ. 2025. Т. 7, № 2(21). С. 48–55. URL: https://doi.org/10.54708/2658-5014-SIIT-2025-no2-p48. EDN XTKJHQ. [[Saitova G. A., Gabdullina E. R. Methodology for Determining Field Projective Cover Based on Remote Monitoring // SIIT. 2025. Vol. 7, No. 2(21). Pp. 48-55. (In Russian).]]

ОБ АВТОРАХ | ABOUT THE AUTHORS

ХАЛИЛОВ Руслан Денисович
Институт механики им. Р. Р. Мавлютова — УФИЦ РАН.
mr.khalilovruslan@gmail.com ORCID: 0009-0009-7562-1702.
Аспирант. Готовит дисс. в обл. системного анализа, управления и обработки информации.

МУСЛИМОВ Тагир Забирович
Институт механики им. Р. Р. Мавлютова — УФИЦ РАН.
tagir.muslimov@gmail.com ORCID: 0000-0002-9264-529X.
И. о. ст. науч. сотр. лаборатории робототехники и управления в технических системах. Магистр прикл. матем. и физ. (Моск. физ.-техн. ин-т, 2012). Канд. техн. наук по сист. анализу, управлению и обработке информации (Уфимск. гос. авиац. техн. ун-т, 2020). Иссл. в обл. управления сложн. техн. объектами, робототехники, систем управления автономными роботами.

KHALILOV Ruslan Denisovich
Mavlyutov Institute of Mechanics — Ufa FR Centre of RAS.
mr.khalilovruslan@gmail.com ORCID: 0009-0009-7562-1702.
Postgraduate Student of the specialty 2.3.1. "System analysis, control and information processing, statistics".

MUSLIMOV Tagir Zabirovich
Mavlyutov Institute of Mechanics — Ufa FR Centre of RAS.
tagir.muslimov@gmail.com ORCID: 0000-0002-9264-529X.
Senior Research Fellow of the Laboratory of Robotics and Control in Technical Systems. Master's degree (Moscow Inst. Phys. & Tech., 2012). Cand. Tech. Sci. (PhD) in system analysis, control, and information processing (Ufa State Aviat. Techn. Univ., 2020). Research in the field of control for complex technical objects, robotics, control systems for autonomous robots.

МЕТАДАННЫЕ | METADATA

Заглавие: Сравнение моделей нейронных сетей для автоматического управления полетом квадрокоптера по заданной траектории

Авторы: Халилов Р. Д., Муслимов Т. З.

Аннотация: Пропорционально-интегрально-дифференциальные (ПИД) регуляторы широко применяются в промышленности и исследовательских задачах благодаря простоте и эффективности. Однако при наличии параметрических неопределенностей и внешних возмущений, особенно в динамически сложных системах вроде квадрокоптеров, остаётся актуальной задача обеспечения их робастности. В работе сравнивается самонастраивающаяся ПИД-схема, использующая подкрепляющее обучение и гибридную нейросетевую архитектуру «актор–критик» для управления ориентацией и высотой полёта квадрокоптера без априорной математической модели, с подобной архитектурой, использующей метод Proximal Policy Optimization (PPO) для оптимизации работы. В обоих случаях коэффициенты усиления регулятора состоят из статической и адаптивной динамической части, при этом обучаются только переменные компоненты. Нейросеть включает два скрытых слоя с сигмоидальными активациями. Обучение проводилось онлайн с оптимизатором ADAM и обратным распространением ошибки, что обеспечивает быструю адаптацию ко внешним возмущениям и изменению массы аппарата. Эксперименты показали высокую

Title: Comparison of neural network models for automatic flight control of a quadcopter along a given trajectory

Authors: Khalilov R. D., Muslimov T. Z.

Abstract: Proportional-integral-differential (PID) regulators are widely used in industry and research tasks due to their simplicity and efficiency. However, in the presence of parametric uncertainties and external disturbances, especially in dynamically complex systems like quadcopters, the task of ensuring their robustness remains urgent. The paper compares a self-adjusting PID scheme using reinforcement learning and a hybrid Actor-Critic neural network architecture to control the orientation and altitude of a quadcopter without an a priori mathematical model, with a similar architecture using the Proximal Policy Optimization (PPO) method for optimization of work. In both cases, the controller gains consist of a static and adaptive dynamic part, while only the variable components are trained. The neural network includes two hidden layers with sigmoid activations. The training was conducted online with the ADAM optimizer and error propagation, which ensures rapid adaptation to external disturbances and changes in the mass of the vehicle. The experiments showed high stability of the systems to mass variations and wind gusts when using trajectories of varying complexity. A comparison of the two methods showed that they did not have a significant difference in deviations from the ideal trajectory; however, the PPO

устойчивость систем к вариациям массы и порывам ветра при использовании траекторий различной сложности. Сравнение двух методов показало, что значительной разницы в отклонениях от идеальной траектории у них нет, однако метод PPO обучался в 2.8 раза быстрее, чем стандартный «актор–критик». Кроме того, метод PPO показал большее отклонение от идеальной высоты при изменении массы дрона в полёте. Результаты подтверждают потенциал гибридных нейросетевых структур для адаптивного управления в условиях неопределённости и рекомендуют разработанный алгоритм к практическому применению в автономных БПЛА, при этом архитектура, использующая стандартную модель «актор–критик», предпочтительнее при изменениях массы квадрокоптера в полёте, а архитектура, использующая PPO – при сложных, длинных маршрутах.

Ключевые слова: Адаптивное ПИД-регулирование; обучение с подкреплением; квадрокоптер; нейросеть; «актор–критик»; самонастраивающийся регулятор; БПЛА

Язык: Русский.

Статья поступила в редакцию 26 сентября 2025 г.

method was trained 2.8 times faster than the standard Actor-Critic. In addition, the PPO method showed a greater deviation from the ideal height when changing the mass of the drone in flight. The results confirm the potential of hybrid neural network structures for adaptive control in conditions of uncertainty and recommend the developed algorithm for practical use in autonomous UAVs. The architecture using the standard Actor-Critic model is preferable for changes in the mass of a quadcopter in flight, and the architecture using PPO for complex, long routes.

Key words: Adaptive PID control; Reinforcement learning; Quadcopter; Neural network; Actor-Critic; Self-adjusting controller; UAV

Language: Russian.

The editors received the article on 26 September 2025.