

УДК 512.6, 517.9, 519.6

## **КВАНТОВЫЙ «СИЛЬНЫЙ» ВЫЧИСЛИТЕЛЬНЫЙ ИИ В ИНТЕЛЛЕКТУАЛЬНОМ УПРАВЛЕНИИ БЕСПИЛОТНЫМ ВОЗДУШНЫМ СУДНОМ МУЛЬТИКОПТЕРНОГО ТИПА: ТЕХНОЛОГИИ ОПТИМИЗАЦИИ БАЗ ЗНАНИЙ НА КВАНТОВЫХ МЯГКИХ ВЫЧИСЛЕНИЯХ**

**Боровинский Виталий Викторович<sup>1</sup>, Капков Роман Юрьевич<sup>2</sup>, Кузьмин Андрей Евгеньевич<sup>3</sup>, Романович Ярослав Сергеевич<sup>4</sup>, Решетников Андрей Геннадьевич<sup>5</sup>, Тятюшкина Ольга Юрьевна<sup>6</sup>, Ульянов Сергей Викторович<sup>7</sup>**

<sup>1</sup>Предприниматель, директор;

ООО «АТОМ»;

Россия, 127566, г. Москва, Высоковольтный пр-д, 21, стр. 5;

e-mail: vitali.borovinski@gmail.com.

<sup>2</sup>Предприниматель, преподаватель-исследователь;

Государственный университет «Дубна»;

Россия, 141980, Московская обл., г. Дубна, ул. Университетская, 19;

e-mail: info@kapkov.pro.

<sup>3</sup>Заместитель руководителя, исследователь;

ООО «АТОМ»;

Россия, 127566, г. Москва, Высоковольтный пр-д, 21, стр. 5;

e-mail: a--kuzmin@yandex.ru.

<sup>4</sup>Студент;

Государственный университет «Дубна»;

141980, Московская обл., г. Дубна, ул. Университетская, 19;

e-mail: yarik23.07.03@gmail.com.

<sup>5</sup>Кандидат технических наук, доцент;

Государственный университет «Дубна»;

Россия, 141980, Московская обл., г. Дубна, ул. Университетская, 19;

Старший научный сотрудник;

Объединенный институт ядерных исследований;

Россия, 141980, Московская обл., г. Дубна, ул. Жолио-Кюри, 6;

e-mail: agreshetnikov@gmail.com.

<sup>6</sup>Кандидат технических наук, доцент;

Государственный университет «Дубна»;

Россия, 141980, Московская обл., г. Дубна, ул. Университетская, 19;

e-mail: tyatyushkina@mail.ru.

<sup>7</sup>Доктор физико-математических наук, профессор;

Государственный университет «Дубна»;

Россия, 141980, Московская обл., г. Дубна, ул. Университетская, 19;

Главный научный сотрудник;

Объединенный институт ядерных исследований;

Россия, 141980, Московская обл., г. Дубна, ул. Жолио-Кюри, 6;

e-mail: ulyanovsv46\_46@mail.ru.

Рассмотрены возможности применения интеллектуальной системы управления (ИСУ) беспилотным воздушным судном (БВС). Подробно представлены этапы проектирования встраиваемой ИСУ на основе технологии квантовых и мягких вычислений. В частности, описывается возможность обучения реального БВС со встроенным генетическим алгоритмом, раскрыты варианты применения математического моделирования для извлечения обучающего сигнала. Особое



Статья находится в открытом доступе и распространяется в соответствии с лицензией Creative Commons «Attribution» («Атрибуция») 4.0 Всемирная (CC BY 4.0) <https://creativecommons.org/licenses/by/4.0/deed.ru>

внимание уделено программным инструментариям – «Оптимизаторам баз знаний на мягких и квантовых вычислениях», которые применяются для проектирования встраиваемой самоорганизующейся квантовой ИСУ в непредвиденных ситуациях управления. Представлены эффекты полученные за счет применения квантовых самоорганизующихся контроллеров, встроенных в контур управления ИСУ. Показано квантовое преимущество и превосходство ИСУ над классическими и нечеткими системами управления.

**Ключевые слова:** интеллектуальное управление, сильный искусственный интеллект, квантовое управление, беспилотное воздушное средство.

**Для цитирования:**

Квантовый «сильный» вычислительный ИИ в интеллектуальном управлении беспилотным воздушным судном мультикоптерного типа: технологии оптимизации баз знаний на квантовых мягких вычислениях / В. В. Боровинский, Ю. Р. Капков, А. Е. Кузьмин [и др.] // Системный анализ в науке и образовании: сетевое научное издание. 2025. № 1. С. 65–123. EDN: JZXJMS. URL: <https://sanse.ru/index.php/sanse/article/view/650>.

**QUANTUM “STRONG” COMPUTATIONAL AI – CONTROL OF UNMANNED AIR VEHICLE OF MULTI-COPTER TYPE: KNOWLEDGEBASE OPTIMIZATIONS ON QUANTUM SOFT COMPUTING TECHNOLOGY**

**Borovinsky Vitaly V.<sup>1</sup>, Kapkov Roman Yu.<sup>2</sup>, Kuzmin Andrey E.<sup>3</sup>,  
Romanovitch Yaroslav S.<sup>4</sup>, Reshetnikov Andrey G.<sup>5</sup>,  
Tyatushkina Olga Yu.<sup>6</sup>, Ulyanov Sergey V.<sup>7</sup>**

<sup>1</sup>*The entrepreneur, Director;  
ATOM LLC;  
21 Vysokovoltny ave., bld. 5, Moscow, 127566, Russia;  
e-mail: vitali.borovinski@gmail.com*

<sup>2</sup>*Businessman, teacher-researcher;  
19 Universitetskaya Str., Dubna, Moscow region, 141980, Russia;  
141980, Dubna, Moscow reg., Universitetskaya str., 19;  
e-mail: info@kapkov.pro*

<sup>3</sup>*The deputy head, researcher;  
ATOM LLC;  
21 Vysokovoltny ave., bld. 5, Moscow, 127566, Russia;  
e-mail: a--kuzmin@yandex.ru.*

<sup>4</sup>*Student;  
Dubna State University;  
19 Universitetskaya Str., Dubna, Moscow region, 141980, Russia;  
e-mail: yarik23.07.03@gmail.com.*

<sup>5</sup>*PhD in Engineering sciences, associate professor;  
Dubna State University;  
19 Universitetskaya Str., Dubna, Moscow region, 141980, Russia;  
Senior Researcher;  
Joint Institute for Nuclear Research;  
6 Joliot-Curie Str., Dubna, Moscow region, 141980, Russia;  
e-mail: agreshetnikov@gmail.com.*

<sup>6</sup>*PhD in Engineering sciences, associate professor;  
Dubna State University;  
19 Universitetskaya Str., Dubna, Moscow region, 141980, Russia;  
e-mail: tyatyushkina@mail.ru.*

<sup>7</sup>*Grand PhD in Physical and Mathematical Sciences, professor;  
Dubna State University;*

19 Universitetskaya Str., Dubna, Moscow region, 141980, Russia;  
 Chief Researcher;  
 Joint Institute for Nuclear Research;  
 6 Joliot-Curie Str., Dubna, Moscow region, 141980, Russia;  
 e-mail: ulyanovsv46\_46@mail.ru.

The possibilities of application an intelligent control system (ISU) for an unmanned aerial vehicle (UAV) are considered. The stages of designing an embedded ICU based on quantum and soft computing technology are presented in detail. In particular, the possibility of training a real UAV with a built-in genetic algorithm is described, and options for using mathematical modeling to extract a training signal are disclosed. Special attention is paid to software tools – "Knowledge base Optimizers based on soft and quantum computing", which are used to design embedded self-organizing quantum information systems in unpredicted control situations. The effects obtained through the use of quantum self-organizing controllers embedded in the control feedback loop of the ICU are presented. The quantum advantage and superiority of ICS over classical and fuzzy control systems is demonstrated.

**Keywords:** intelligent control, strong artificial intelligence, quantum control, unmanned aerial vehicle.

### For citation:

Borovinsky Vitalii V., et al. Quantum “strong” computational AI – control of unmanned air vehicle of multi-copter type: knowledgebase optimizations on quantum soft computing technology. *System analysis in science and education*, 2025;(1):65-123 (in Russ). EDN: JZXJMS. Available from: <https://sanse.ru/index.php/sanse/article/view/650>.

## Введение

Понятие беспилотное воздушное судно (БВС) содержится в Воздушном кодексе Российской Федерации, под которым определяется воздушное судно, управляемое, контролируемое в полете пилотом, находящимся вне борта такого воздушного судна (внешнего пилота) [1]. Современная эпоха БВС охватила широкий спектр общественных отношений [2]. Они используются в различных отраслях, таких как сельское хозяйство, видеосъемки, мониторинг, образование, аэрологистика и иных сферах деятельности (см. рис. 1) [3].

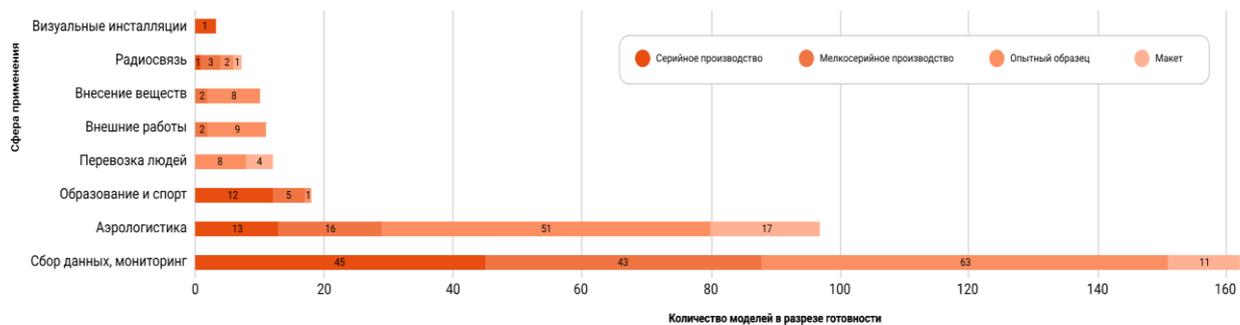


Рис. 1. Сферы применения БВС

Число используемых БВС ежегодно растет. Сравнивая данные о зарегистрированных гражданских воздушных судах на момент 2023 и 2024 годов, видна тенденция роста и заинтересованности в данной сфере. На начало 2023 года было зарегистрировано 70201 устройств, на начало 2024 – 93107 штук [2], прирост составил более 30%. Актуальность вопроса автоматического управления БВС возрастает. Рынок имеет положительный рост на протяжении последних 10 лет, следовательно, возникает спрос на автоматическое управления [4], в том числе и в различных условиях. В условиях (порой резко изменяющихся) природных воздействий повышается необходимость в точности и робастности встраиваемой бортовой системы управления.

Рассматривая альтернативы решения задач [6, 7], связанных с повышением устойчивости БВС во время полета, основной путь решения – интеллектуализация системы управления (ИСУ). На протяжении довольно продолжительного времени крупные разработчики модернизируют свои программы систем управления для достижения точного и робастного управления в широком диапазоне

условиях. Однако, такие подходы обязывают инженеров проводить дополнительные подстройки параметров системы управления, что с одной стороны вносит субъективизм в процесс настройки и эксплуатации БВС, с другой стороны накладывает ответственность на этот сложный процесс, имеющий в себе довольно много проблем и особенностей.

## 1. Описание модели БВС

Рассмотрим некоторую классификацию БВС. В табл. 1 представлена классификация БВС по взлетной массе, взятая из [8].

Таблица 1. Классификация беспилотных воздушных судов по взлетной массе

Класс БВС	Взлетная масса, кг	Заявленная дальность действия, км	Примеры
Тяжелые БВС	>400	70–300	“Орион”, “Фортпост”
Средние БВС	10–50	40–70	<i>Zala Aero</i> , “Орлан”
Легкие БВС средней дальности	до 10	до 15	“Шмель”, <i>DJI Matrice 300</i>
Легкие БВС малой дальности	до 1	до 7	<i>DJI Mavic 2</i> , <i>DJI Mavic 3T</i>
Микро– и мини– БВС	< 0,3	до 30	<i>DJI Mavic Mini 2</i>

Также все виды БВС делятся по типу строения (рис. 2):

1. Самолетного типа – принцип работы заключается в использовании крыла для создания подъемной силы. Они не могут зависать в воздухе, однако это компенсируется дальностью полета.
2. Вертолетного типа – принцип работы заключается в использовании нескольких бесколлекторных двигателей с пропеллерами. За счет изменения силовой тяги на каждом роторе аппарат имеет возможность полета в любой плоскости, однако радиус действия сильно ограничен.
3. Гибридного типа – принцип работы заключается в совмещении строений предшественников для возможности парения над точкой и более долгого времени полета.



Рис. 2. БВС самолетного, вертолетного и гибридного типа

В данной работе в качестве примера применяемого БВС, будем рассматривать классическую модель гексакоптера «S550». На рис. 3 и 4 представлены модель БВС и оси вращения кинематики движения БВС, соответственно.

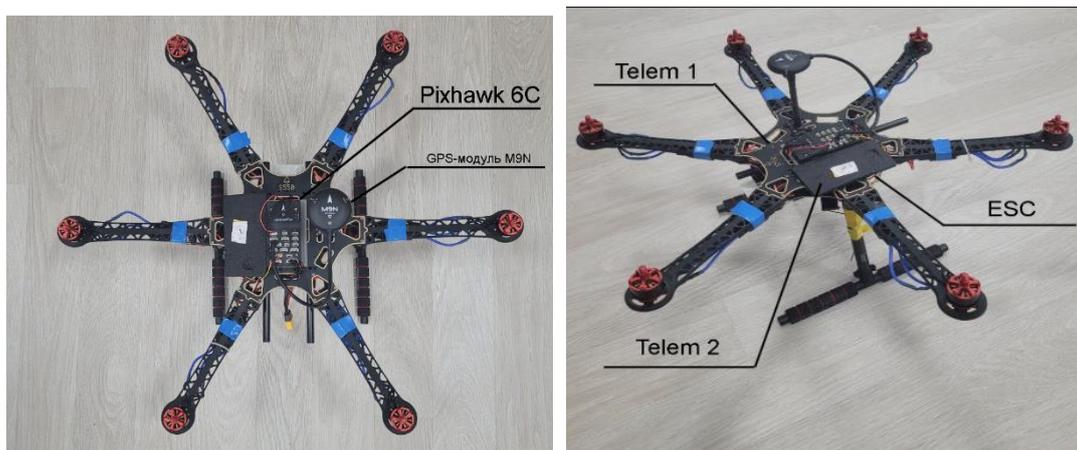


Рис. 3. Макет БВС

В основе элементной базы БВС на борту установлены:

1. Полетный контроллер *Pixhawk6C*, реализующий связь всех периферийных устройств и компонентов БВС, а также являющийся центральным устройством для выполнения заложенной в него программы;
2. *GPS* модуль *M9N* – предназначен для *GPS*–связи, требуемой для обучения и тестирования элементов интеллектуального управления;
3. Два устройства телеметрии: с частотой 915 и 433 *Mhz* – используются для связи с наземной станцией управления и обмена данными между обучающей программой и БВС;
4. Используемые литий–полимерные батареи: 3S и 4S на 5000 и 8000 *mAh* соответственно;
5. Контроллеры двигателей *ESC*: *esc 25A*
6. Двигатели: 2212/92.

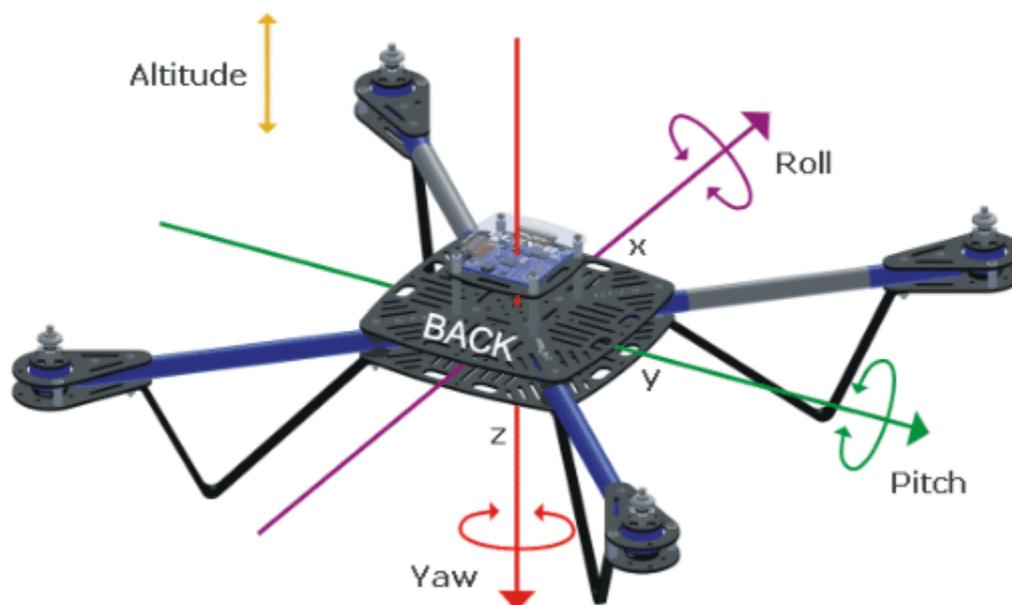


Рис. 4. Оси вращения Roll, Pitch, Yaw

Главным элементом бортовой аппаратуры БВС является полетный контроллер, который принимает и обрабатывает команды от пункта управления или бортового компьютера и транслирует их в двигатели и другие элементы бортового комплекса [5]. Также на борту присутствуют различные стандартные авиационные датчики:

1. барометр – определяет высоту полета;
2. акселерометр – измеряет ускорение БВС;
3. гироскоп – определяет положение БВС в пространстве;
4. гироскоп – указывает направление полета.

Помимо этого, в состав бортового комплекса БВС входят модули спутниковой системы навигации, приемопередатчики системы связи и т.д.

Рассмотрим программное обеспечение системы управления БВС. Существует множество различных программ (далее прошивок), для реализации системы управления БВС [9]. Один из популярных вариантов реализации – прошивка *ArduCopter* [10], являющаяся одной из частей *ArduPilot*, взятая за основу. Ниже (см. рис. 5) приведена высокоуровневая диаграмма системы управления с ПИД регулятора от приема сигнала управления до полной его обработки и передачи на моторы.

ArduCopter V4.X STABILIZE Roll, Pitch & Yaw PID's

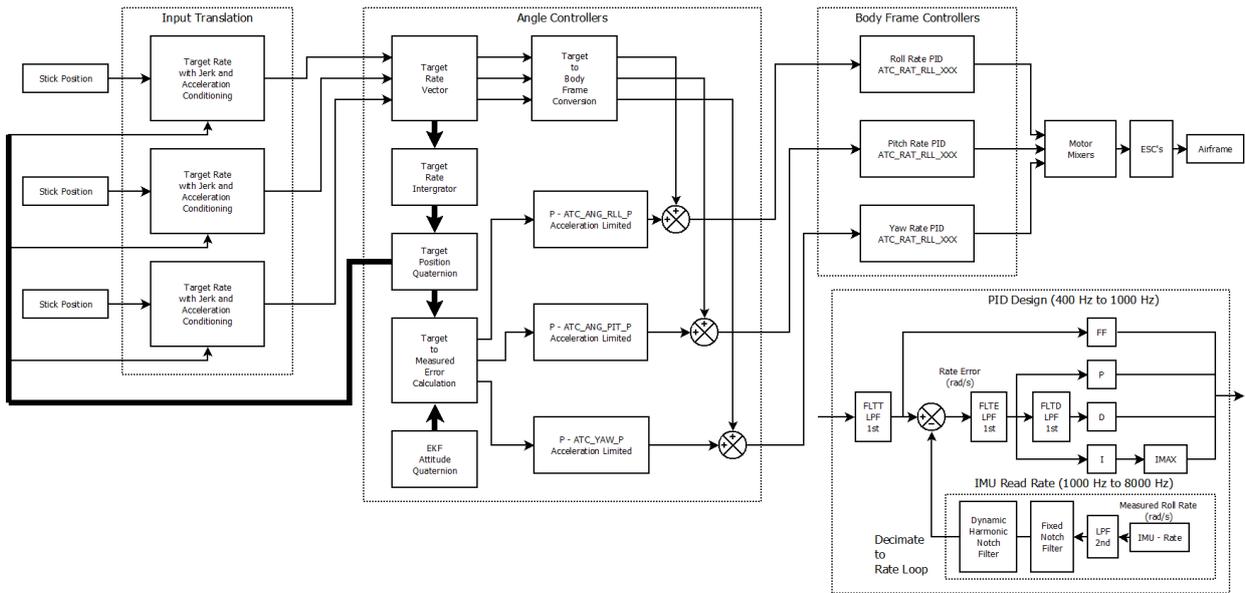


Рис. 5. Диаграмма управления БВС

Рассмотрим отдельно работу регуляторов для каждой оси. *P*-контроллер получает на вход требуемый угол. Далее, оценивая актуальные угол и скорость получает необходимую информацию для контроллеров по изменению параметров для достижения требуемого условия. Контроллер с квадратным корнем (см. рис. 6) гарантирует плавность движения [11]. Достигается это за счет планирования выигрышной позиции, основанной на ошибке текущей позиции с формой кривой усиления, определяемой максимально допустимым ускорением.

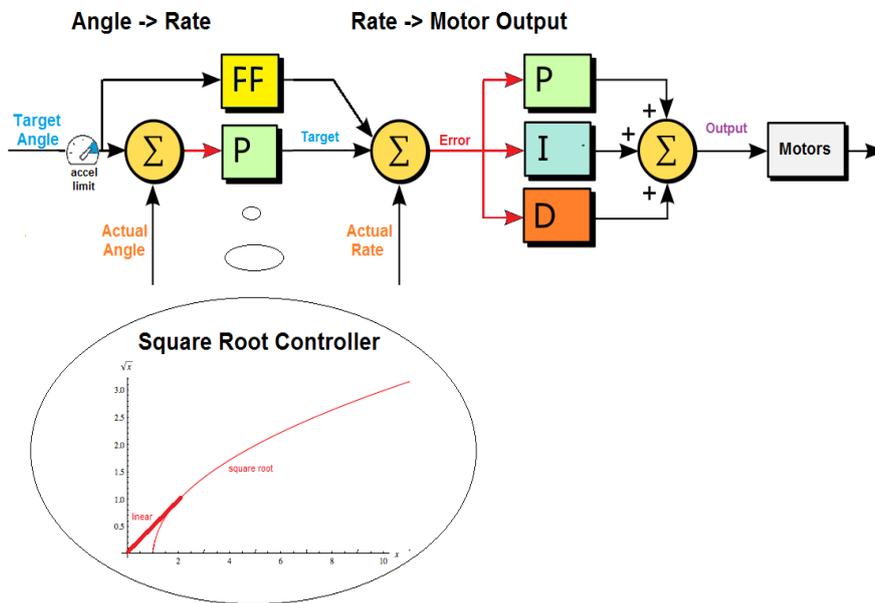


Рис. 6. Пример работы системы управления от заданного угла до мотора

На приведенной ниже диаграмме (см. рис. 7) показан путь следования кода от пилотного ввода до шим-выхода [11].

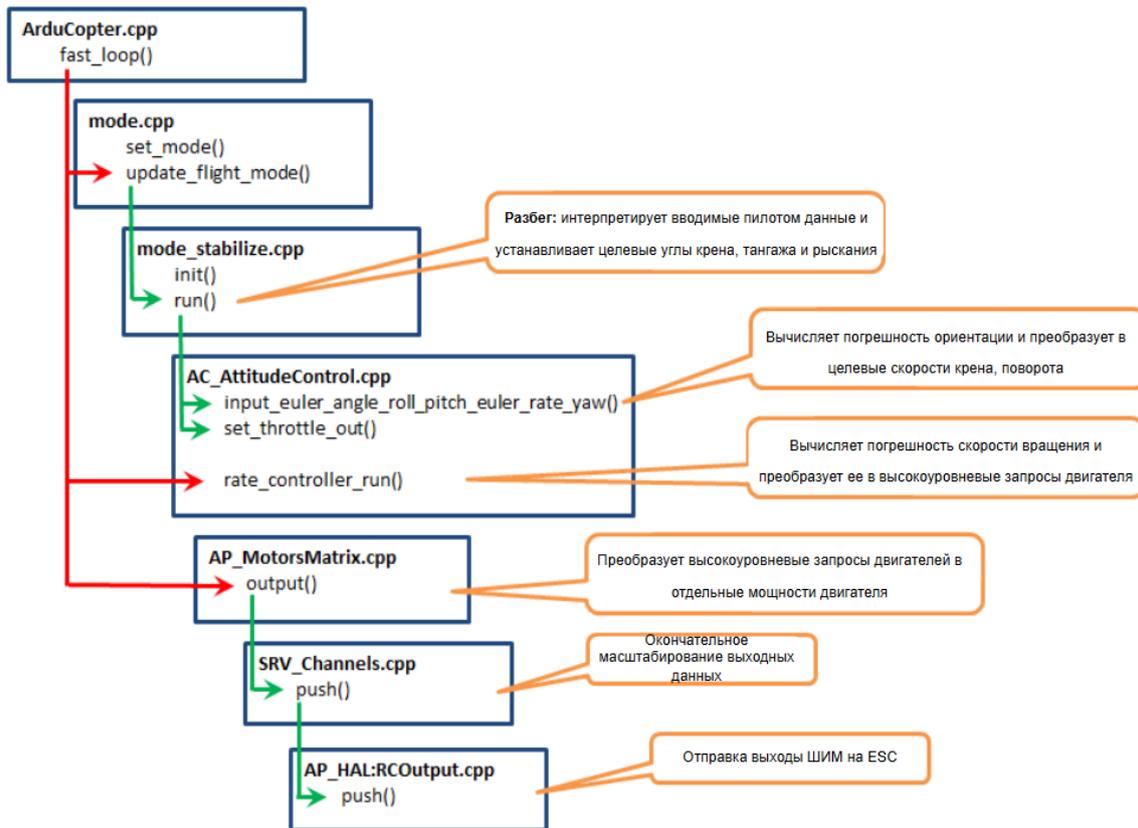


Рис. 7. Взаимодействие библиотек ArduPilot

При каждом обновлении (например, 400 Гц на *Pixhawk*, 100 Гц на *APM2.x*) происходит следующее:

- Вызывается функция верхнего уровня *flight-mode.cpp* “*update\_flight\_mode()*”. Эта функция проверяет режим полета транспортного средства (т.е. переменная “*control\_mode*”), а затем вызывает соответствующую функцию <режим полета>\_run() (т.е. *stabilize\_run* для режима стабилизации, *rtl\_run* для режима *RTL* и т.д.). Функцию <режим полета>\_run() можно найти в файле с соответствующим именем *mode\_<режим полета>.cpp* (например, *mode\_stabilize.cpp*, *mode\_rtl.cpp* и т.д.).
- Функция <режим полета>\_run отвечает за преобразование введенных пользователем данных (найденных в *g.rc\_1.control\_in*, *g.rc\_2.control\_in* и т.д.) в угол наклона, скорость поворота, скорость набора высоты и т.д., которые подходят для данного режима полета. Например, *AltHold* преобразует данные пользователя о крене и тангаже в углы наклона (в градусах), данные о рыскании преобразуются в скорость вращения (в градусах в секунду), а данные о дроссельной заслонке преобразуются в скорость набора высоты (в см/с).
- Последнее, что должна сделать функция <режим полета>\_run, это передать данные о желаемых углах, скорости и т.д. в библиотеки управления ориентацией и/или положением (обе они хранятся в папке *AC\_AttitudeControl*).
- Библиотека *AC\_AttitudeControl* предоставляет пять возможных способов управления положением транспортного средства, три наиболее распространенных из которых описаны ниже.
- *Angle\_ef\_roll\_pitch\_rate\_ef\_yaw()*: этот параметр принимает угол “земной оси” для крена и тангажа и скорость “земной оси” для рыскания. Например, использование этой функции при крене = -1000, тангаже = -1500, рыскании = 500 означает, что БВС наклоняется влево на 10 градусов, наклоняется вперед на 15 градусов и поворачивает вправо со скоростью 5 градусов в секунду.
- *Angle\_ef\_roll\_pitch\_yaw()*: этот параметр принимает значения углов “земной оси” для крена, тангажа и рыскания. аналогично описанному выше, за исключением того, что для обеспечения поворота на 5 градусов означает поворот транспортного средства на 5 градусов к востоку от северного направления.
- *Rate\_bf\_roll\_pitch\_yaw()* : это значение принимает частоту вращения корпуса (в градусах в секунду) для тангажа крена и рыскания. Например, использование этой функции при крене = -

1000, тангаже =  $-1500$ , рыскании =  $500$  приведет к тому, что транспортное средство будет поворачиваться влево со скоростью  $10$  градусов в секунду, наклоняться вперед со скоростью  $15$  градусов в секунду и поворачиваться вокруг оси  $z$  со скоростью  $5$  градусов в секунду.

- После выполнения любых вызовов этих функций вызывается функция `AC_AttitudeControl::rate_controller_run()`. Это преобразует выходные данные из перечисленных выше методов во входные данные по крену, тангажу и рысканию, которые отправляются в библиотеку `AP_Motors` с помощью методов `set_roll`, `set_pitch`, `set_yaw` и `set_throttle`.

- Библиотека `AC_PosControl` позволяет управлять положением автомобиля в  $3D$ -пространстве. Обычно используются только более простые методы по оси  $Z$  (например, управление высотой), поскольку более сложные режимы трехмерного позиционирования (например, зависание) используют библиотеку `AC_WPNav`. В любом случае, некоторые часто используемые методы этой библиотеки включают:

- `Set_alt_target_from_climb_rate()` : принимает скорость набора высоты в см/с и обновляет заданную абсолютную высоту

- `Set_pos_target()` : принимает  $3D$ -вектор положения, который представляет собой смещение от исходного положения в сантиметрах. Если вызываются какие-либо методы в `AC_PosControl`, то код режима полета также должен вызывать метод `AC_PosControl::update_z_controller()`. Это запустит  $PID$ -контур управления положением по оси  $z$  и отправит данные о низком уровне дроссельной заслонки в библиотеку `AP_Motors`. Если вызываются какие-либо методы по осям  $x, y$ , то необходимо вызвать `AC_PosControl::update_xy_controller()`.

- Библиотека `AP_Motors` содержит код “смешивания двигателей”. Этот код отвечает за преобразование значений крена, тангажа, рыскания и дроссельной заслонки, полученных из библиотек `AC_AttitudeControl` и `AC_PosControl`, в абсолютные выходные данные двигателя (т.е. значения ШИМ). Таким образом, библиотеки более высокого уровня будут использовать эти функции:

- `Set_roll()`, `set_pitch()`, `set_yaw()`: принимают значения крена, тангажа и рыскания в диапазоне от  $-4500$  до  $4500$ . Это не желаемые углы и даже не скорости, а скорее просто значение. Например, значение `set_roll(-4500)` означает, что нужно выполнить бросок влево как можно быстрее.

- `Set_throttle()`: принимает абсолютное значение дроссельной заслонки в диапазоне от  $0$  до  $1000$ .  $0$  = двигатели выключены,  $1000$  = полный газ.

- Существуют разные классы для каждого типа рамы (квадроцикл,  $Y6$ , традиционный вертолет), но в каждом из них есть функция “`output_armed`”, которая отвечает за преобразование этих значений крена, тангажа, рыскания и дроссельной заслонки в шим-выходы. Это преобразование часто включает в себя внедрение “исправления стабильности”, которое определяет приоритет одной оси управления над другой, когда входные запросы выходят за физические пределы рамы (например, максимальная скорость и максимальный крен невозможны для квадроцикла, потому что некоторые двигатели должны быть меньше, чем другие, чтобы вызвать крен). В нижней части функции “`output_armed`” есть вызов функции `hal.rcout->write()`, которая передает желаемые шим значения на уровень `AP_HAL`.

- Библиотеки `AP_HAL` (уровень аппаратной абстракции) обеспечивают единый интерфейс для всех плат. В частности, функция `hal.rc_out_write()` приведет к появлению указанного  $PWM$ -сигнала, полученного из класса `AP_Motors`, на соответствующем выводе `pin` для платы.

Таким образом реализована система управления БВС на базе семейства программ `Ardupilot`. Далее рассмотрим возможность применения технологий мягких и квантовых вычислений в задаче настройки системы управления и её дальнейшее применение.

## 2. Интеллектуальное управление на основе мягких вычислений

В рассматриваемой задаче используем третье поколение ИСУ (рис. 8) [12–14], в котором используется так называемая *интеллектуальная глобальная обратная связь* (ИГОС). Её применение дает возможность извлекать объективные знания непосредственно из самого динамического поведения ОУ и исполнительного устройства системы автоматического регулирования. Контур ИГОС включает ГА для извлечения обучающего сигнала и информации об оптимальном сигнале управления (исходя из динамического и термодинамического поведения самого ОУ и ПИД-регулятора), и нечёткую

нейронную сеть, аппроксимирующей данный оптимальный сигнал управления с помощью заданной структуры нейронной сети используемую в качестве встраиваемой базы знаний нечёткого регулятора. Основным блоком в структуре ИСУ является система моделирования оптимального сигнала управления (СМОСУ) с помощью ГА и выбранного критерия качества управления, заданного в виде одной из составляющих вектор–функции пригодности ГА.

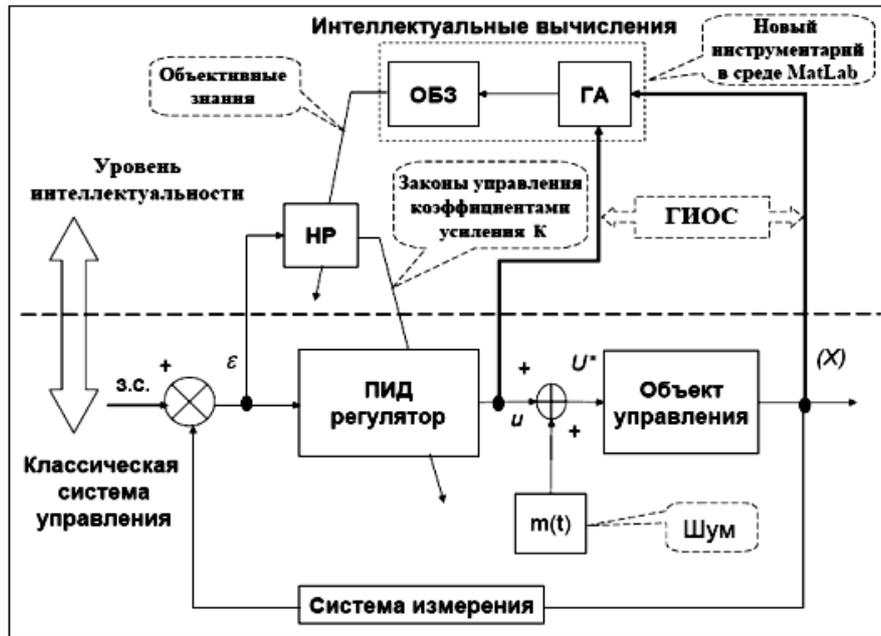


Рис. 8. Типовая структура ИСУ третьего поколения

Выходом СМОСУ на рис. 8 является ОС (оптимального управления) в виде, следующих входных/выходных данных:  $\{E(t_i), K(t_i)\}$ ,  $i = 1, \dots, n$ , где

$E(t_i) = \{e(t_i), \dot{e}(t_i), \int e(t_i) dt_i\}$  – вектор, компонентами которого являются ошибка управления, ее производная и интеграл ошибки соответственно;  $K(t_i) = \{k_p(t_i), k_d(t_i), k_i(t_i)\}$  – оптимальные (с точки зрения заданной функции пригодности ГА) параметры ПИД–регулятора;  $t_i$  – момент времени.

Основной проблемой, решаемой ИСУ третьего поколения с ИГОС, является наличие возможности достижения требуемого уровня робастности на заданном классе как параметрических, так и внешних случайных возмущений различной вероятностной природы (с различными функциями плотности распределения вероятностей).

Другой важной проблемой является определение требуемого соотношения между точностью описания (аппроксимации) ОС и необходимым уровнем робастности всей структуры ННС.

Обе указанные проблемы решаются на втором этапе технологии построения БЗ ИСУ с помощью программных средств инструментария, названного *Оптимизатор Баз Знаний* (ОБЗ).

Рассмотрим далее более подробно саму технологию проектирования встраиваемого интеллектуального управления.

## 2.1 Генетический алгоритм в контуре управления БВС

Рассмотрим кратко основы ГА, как одного из наиболее популярных алгоритмов роевого искусственного интеллекта, или роевой оптимизации. Генетический алгоритм (ГА) — это эвристический алгоритм поиска, основанный на понятиях естественного отбора и генетики. ГА — это подмножество гораздо большей ветви вычислений, известной как эволюционные вычисления и роевой

искусственный интеллект. В ГА (см. рис. 9) выполняется поиск глобально хорошего решения из заданного пространства поиска.

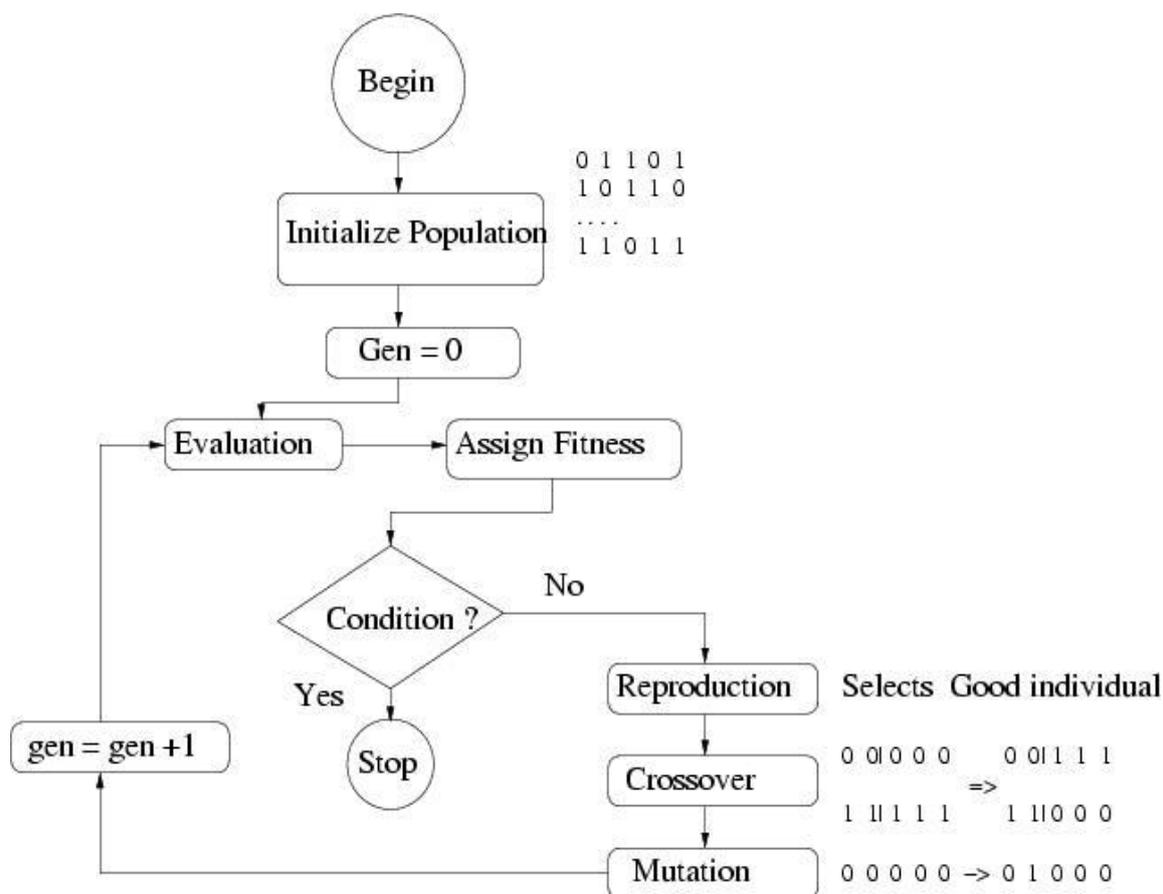


Рис. 9. Генетический алгоритм.

Структурно ГА работает со свойствами самой функции и не требует знаний о поведении ее производных. В процессе выполнения алгоритма, решения подвергаются рекомбинации и мутации (по аналогии естественной эволюции), порождая новые решения. Процесс повторяется заданное число поколений. Каждому подходящему решению присваивается значение его оценки (пригодности) в зависимости от заданной экспертом функции. Наиболее подходящие решения получают высокий шанс перейти в следующие поколения с применением операторов скрещивания и мутации.

У ГА есть различные преимущества:

1. Быстрота и эффективность по сравнению с традиционными методами для большого пространства поиска;
2. Возможность распараллеливания;
3. Оптимизация как непрерывных, так и дискретных функций, а также многоцелевые и многокритериальные задачи;
4. Оптимизация большого количества параметров;
5. Оператор мутации позволяет избегать локальных экстремумов.

Возможны различные варианты применения ГА в задаче обучения БВС. Однако их можно разделить на две большие группы:

1). Использование ГА совместно с математическим моделированием БВС, в частности оценки качества решений. Данный подход обладает очевидными преимуществами при моделировании сложных ситуаций управления и получения оптимального решения. Кроме того, в качестве искомого решения могут выступать оптимальные траектории изменения коэффициентов регулятором системы управления.

2). Второй подход предполагает настройку реального БВС с внедрением ГА непосредственно в систему управления, что позволяет учитывать особенности самого объекта управления.

Далее рассмотрим оба варианта. Здесь, коротко, сформулируем задачу и пространство поиска для генетического алгоритма. В случае применения ГА для реального объекта управления предполагается настройка параметров ПИД регулятора. Соответственно диапазон изменения этих параметров и будет пространством поиска. Что касается функции пригодности, то тут много зависит от свойств обучаемой (проверяемой) траектории, которая будет использоваться для оценки каждого решения. Например, в случае вертикального зависания на точке, можно использовать только значение отклонения по углам в качестве оценки. В случае более сложной траектории и функция оценки будет изменяться.

Для начала рассмотрим возможность использования моделирования, для интеллектуализации БВС.

Для того, чтобы использовать генетический алгоритм при обучении как модели, так и реального объекта, применяется язык программирования *Python*. Далее устанавливаются библиотеки ГА и доработанная *MAVLink*. В зависимости от операционной системы требуется настроить порты соединения, с которыми работает ГА (в случае с *Linux* это *USB0*, с *Windows* – *COM*–порт, в случае с симуляцией настройка портов не требуется). Для этого, также могут понадобиться дополнительные библиотеки. Далее изменяем файл *comconfig.txt* и *mavcomm.py* в соответствии с портами.

Первоначально требуется загрузить так называемый *\*.lua* скрипт на борт. Делается это посредством распространенного программного обеспечения *mission planner* [1]. Запускаем БВС и поднимаем на высоту не менее пяти метров. Далее для запуска ГА на реальном объекте требуется запустить файл *ga\_time\_check.py* и переставить значение *Scripting 1* в положение *MID* через *mission planner* или тумблером на пульте управления. Таким образом включить скрипт обучения. Далее процесс выполняется автоматически. При замене батареи генетический алгоритм находится в режиме ожидания. При повторном подключении процесс продолжится. При окончании ГА БВС зависает в режиме полного удержания позиции.

Симуляция предусмотрена для систем *Linux*. Требуется запустить файл *ga\_sim\_time\_check.py*. В данной программе предусмотрен запуск *Gazebo* и *Ardupilot*. Далее программа выполняется автоматически, по предварительно загруженному скрипту *.lua*.

## 2.2. Программное обеспечение для работы в среде моделирования БВС с применением генетического алгоритма

Первичная установка требуемого программного стека предполагает наличие *Gazebo* [6] (версия *Harmonic*) и *Ardupilot SITL* [4] (см. рис. 10).

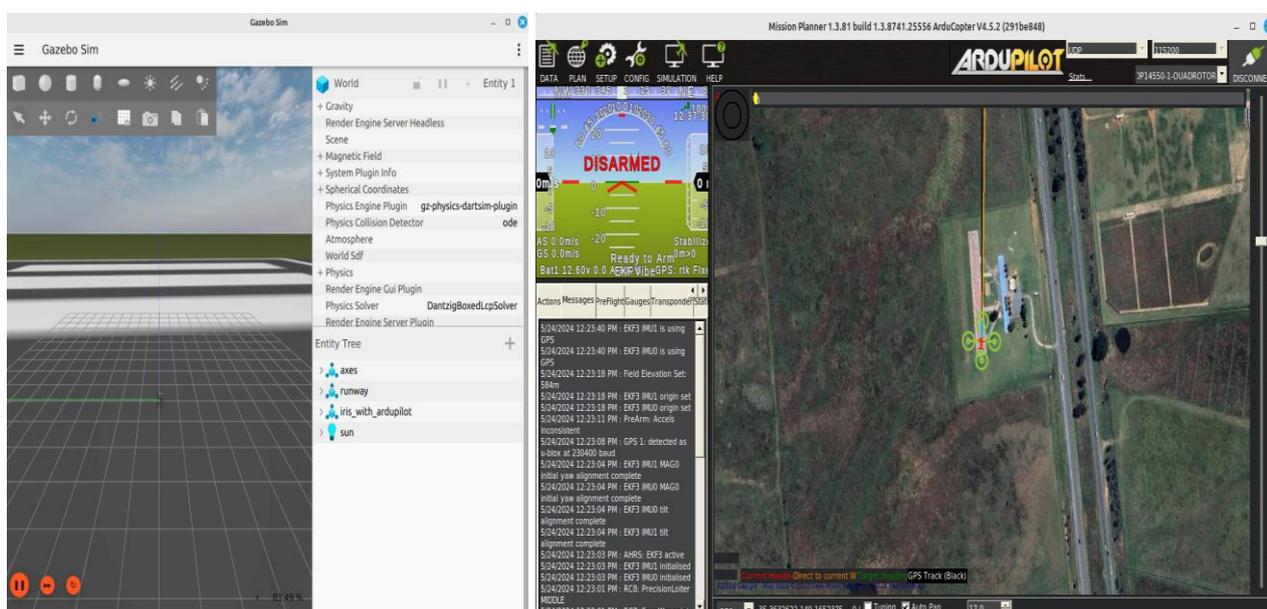


Рис. 10. Связь Gazebo и Mission Planner

Связь между *Gazebo* и *Ardupilot* происходит по *TCP*-порту 5760 (*Serial 0*). При запуске оба приложения должны быть в рабочем состоянии, так как *Ardupilot* принимает *JSON*-файлы от *Gazebo* (см. рис. 11).

Также для анализа результатов и задания миссий полета используется *Mission Planner*. Связь формируется через *TCP*-порт 5762 (*Serial 1*) к *Ardupilot*.

Для запуска симуляции выполняются следующие этапы:

1. Запуск модели *iris-runway* в *Gazebo*);
2. Запуск прошивки *Ardupilot*;
3. Запуск *Mission Planner*.

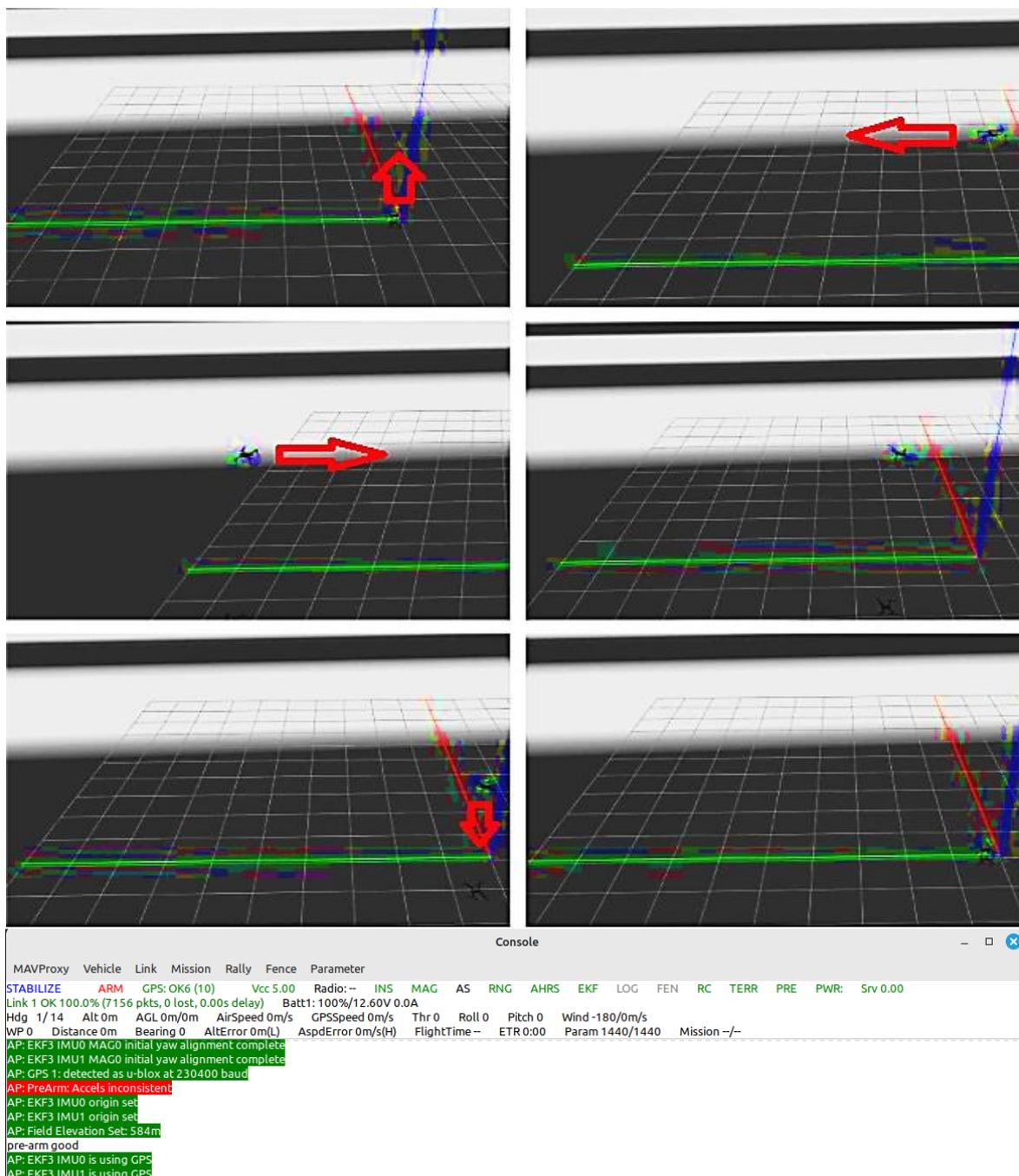


Рис. 11. Выполнение задания в Gazebo

Настройка крена, тангажа и рыскания была рассчитана на 1800 итераций, поделенных на 9 поколений. Был задан маршрут по квадрату, который задействует все системы стабилизации. Достижимая скорость 6 м/с на расстояние 30 метров в каждом направлении. На симуляцию потребовалось около 20 часов, в среднем одна итерация занимала 40 секунд. Также использовался интервальный генетический алгоритм, который в процессе решения одного полета проверял 10 различных наборов параметров, заменяя их в соответствии с частотой системы. Для оценки качества была использована квадратичная интегральная ошибка по осям крена, тангажа и рыскания. Далее представлены примеры оценки сходимости каждого параметра на определенном наборе параметров (группе) и уменьшение ошибки системы, где первые три графика – *PID* – параметры крена, следующие 3 – *PID*–параметры тангажа и последние 3 – *PID*–параметры рыскания (рис. 12 – 14).

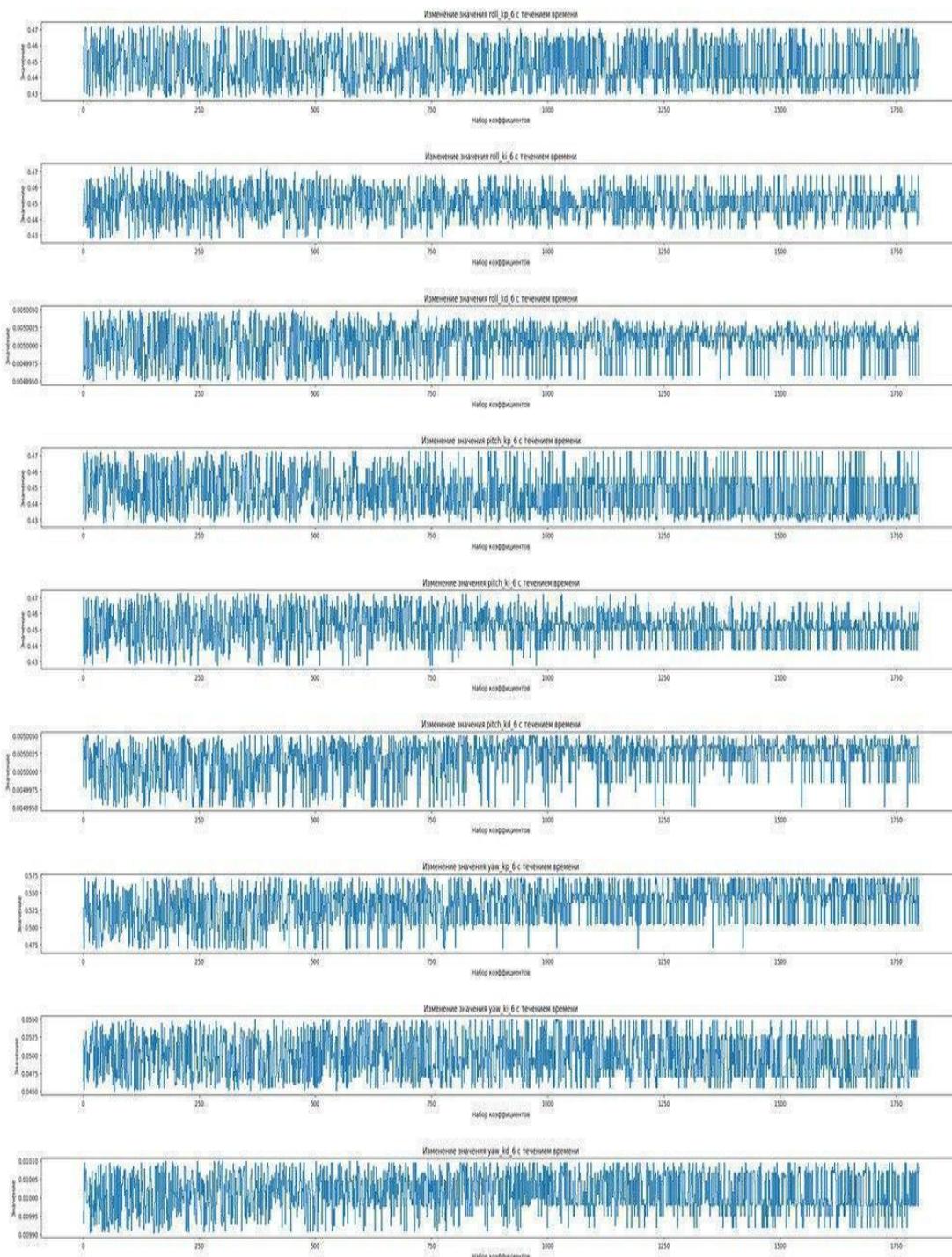


Рис. 12. Сходимость пропорционального, интегрального и дифференциального параметров крена, тангажа и рыскания в 7 группе

Результаты, представленные на рис. 13 и 14, демонстрируют сходимость генетического алгоритма по поколениям и итерациям.

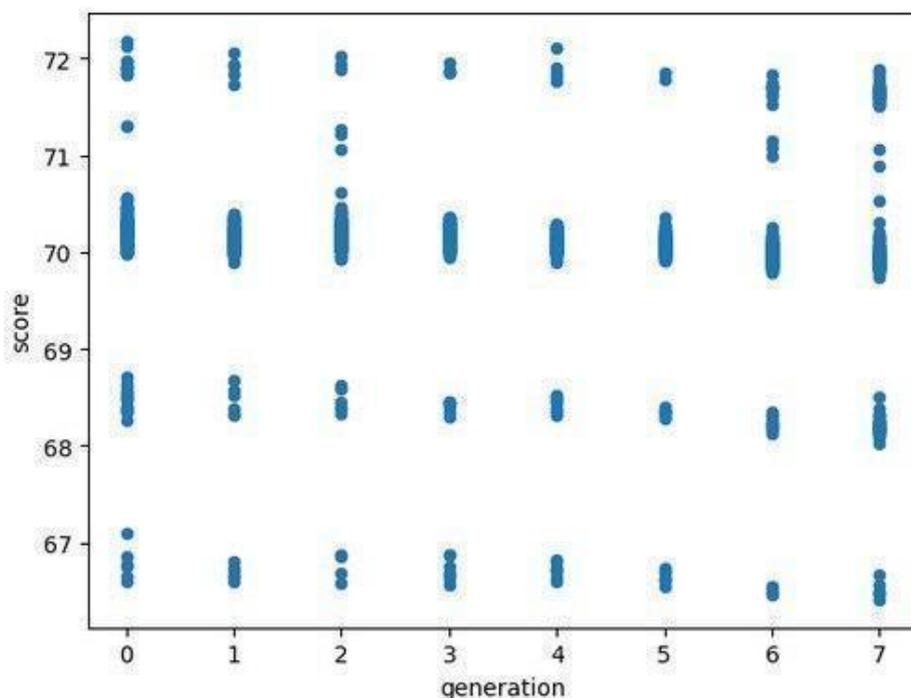


Рис. 13. Сходимость генетического алгоритма по поколениям

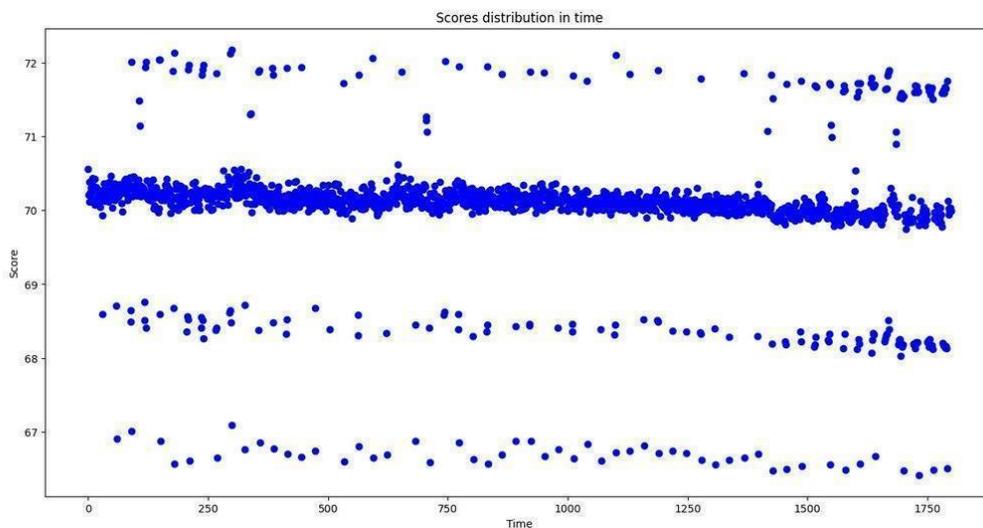


Рис. 14. Сходимость генетического алгоритма по итерациям

Можно заметить, что с течением времени и в процессе математического моделирования, оценка погрешности решений уменьшается, что свидетельствует о сходимости ГА к оптимальным параметрам регулятором, с точки зрения выбранного критерия качества.

### 2.3. Применение генетического алгоритма на реальном объекте

В общем виде схема алгоритма САУ с ГА имеет следующий вид (Рисунок 15).

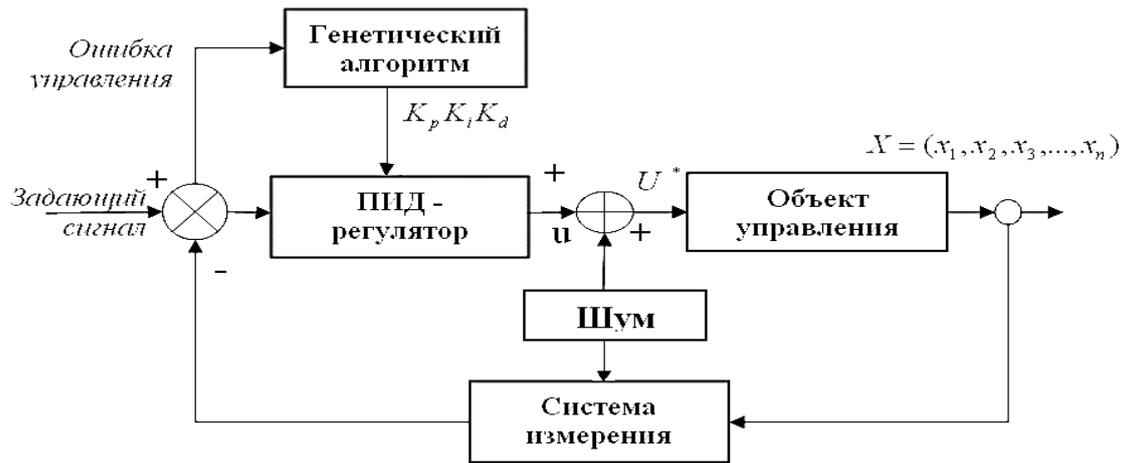


Рис. 15. Блок-схема САУ с системой адаптации

Применительно к настройке ПИД-регуляторов БВС, каждое решение представляет значения коэффициентов усиления, сложенные в одну хромосому. Начальная популяция может быть получена случайным образом или путем экспертной оценки из предварительно проведенных экспериментов. Так как операции скрещивания и мутации работают с закодированными параметрами, коэффициенты кодируются в бинарную форму с заданной точностью, соответствующей длине бинарной строки. Также определяются диапазоны поиска коэффициентов. Применительно к данной системе диапазоны поиска коэффициентов равны  $k_p[0,42; 0,46]$ ,  $k_i[0,42; 0,46]$ ,  $k_d[0,005; 0,01]$  для параметров крена и тангажа. Для настройки коэффициентов усиления ПИД-регуляторов используем следующую функцию пригодности:

$$f(x_r) = \frac{1}{T} \frac{1}{1 + t \cdot \int_0^t e^2 dt}$$

где  $t$  – момент времени моделирования и эксперимента,  $T$  – время проверки одного решения,  $e$  – угол отклонения. Диапазоном интегрирования является период проверки одного решения.

В этом случае большим значениям функции пригодности соответствуют малые значения ошибки управления и, следовательно, лучшее качество управления ПИД-регулятора.

Настройка ПИД-регуляторов с использованием модели и генетического алгоритма не должно вызывать трудностей. Однако применительно к реальному ОУ следует уделить особое внимание выбору временного периода оценки решения и выбирать его, исходя из динамических и информационных свойств работы контроллера и системы в целом.

На рис. 16 представлен алгоритм настройки ГА – ПИД-регулятора для БВС.

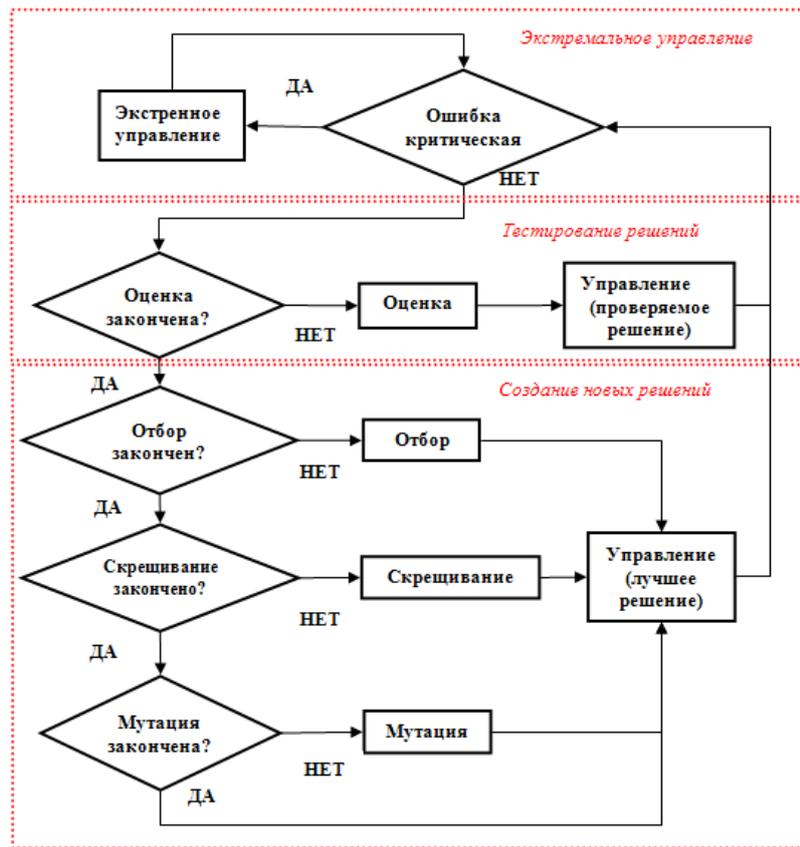


Рис. 16. Алгоритм адаптации ГА – ПИД-регулятора

После каждой итерации БВС управляется с использованием найденного лучшего решения. Это позволяет ГА продолжать работу, не потеряв управления макетом.

Введение экстремального управления осуществляется в момент достижения величины отклонения определенного заранее порога. В моменты работы операторов селекции, скрещивания и мутации, процессорное время занято обработкой данных и нахождением решений, а управляющее воздействие подсчитывается при лучших коэффициентах  $k_P, k_D, k_I$ . Графики изменения коэффициентов такой системы управления представлены далее.

Для проведения обучения системы управления на реальном объекте БВС была разработана схема из серии и последовательности полетов. В таблице 2 представлена последовательности и временной ресурс необходимый для проведения такой работы.

Табл. 2. Программа настроечных полетов

Наименование полета	Временной ресурс (полетные часы)
Настройка классическим способом (определение допустимых диапазонов параметров управления)	1 ч
Настройка удержания позиции (в ситуации 1)	1 ч
Настройка движения вперед – назад (в ситуации 1)	1 ч
Настройка движения по кругу с удержанием курса в центр (в ситуации 1)	1 ч
Построение нечёткого контроллера для первой ситуации (яу питч ролл)	24 ч
Проверка качества управления нечёткого регулятора для ситуации 1	1 ч

Настройка удержания позиции (в ситуации 2)	1 ч
Настройка движения вперед – назад (в ситуации 2)	1 ч
Настройка движения по кругу с удержанием курса в центр (в ситуации 2)	1 ч
Построение нечёткого контроллера для второй ситуации (яу питч ролл)	24 ч
Испытательные полёты на НР1 и НР2 в различных ситуациях	2 ч (в двух ситуациях)
Проектирование квантового контроллера	24 ч
Испытательные полеты на квантах (проверка корреляций)	4 ч (в двух ситуациях)
<b>ИТОГО</b>	<b>62 полетных часа</b>

На реальном объекте использовался генетический алгоритм в режиме реального времени. Проверялись 135 решений, имеющих 9 поколений. В качестве маршрута проверки одного решения использовался полет по квадрату, скорость перемещения равна 6 м/с, каждую сторону БВС пролетал по 6 сек. Был также использован интервальный метод подбора параметров, для этого применялись 4 группы, по одной на каждую сторону. Производилась настройка ПИД регуляторов крена, тангажа, рысканья (см. рис. 17).

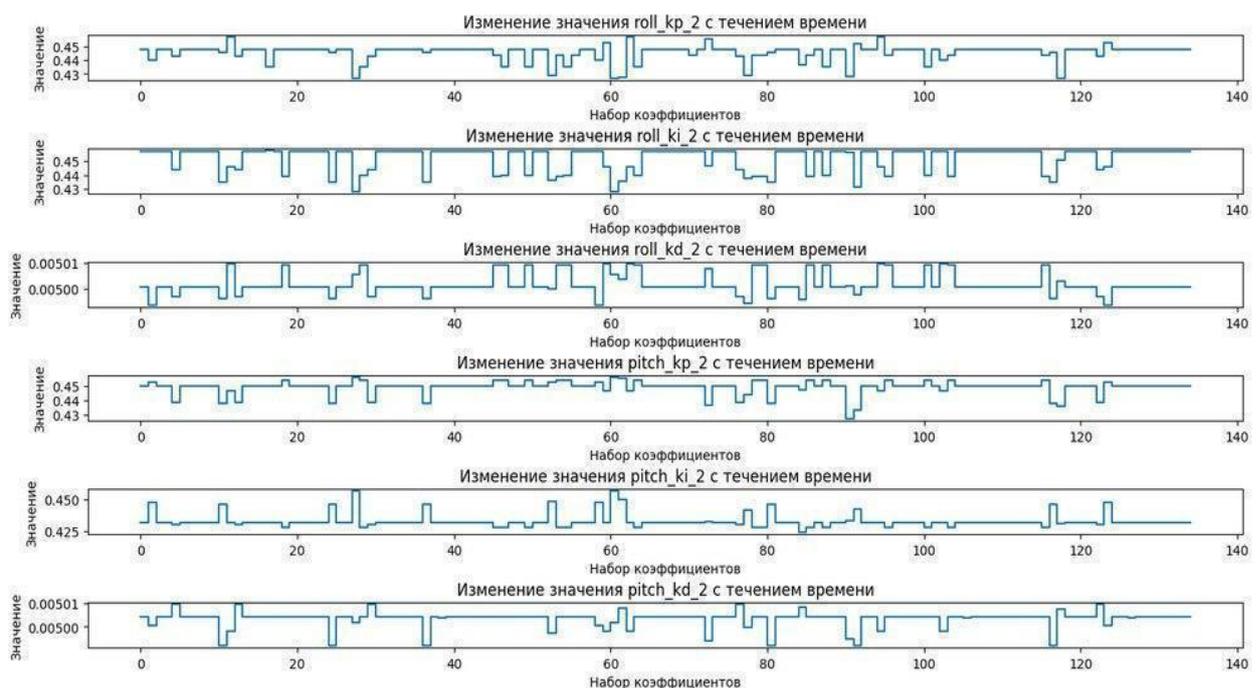


Рис. 17. Сходимость пропорционального, интегрального и дифференциального параметров крена и тангажа в группе

Далее представлены сходимости каждого параметра на определенном наборе параметров (группе) и изменение функции пригодности (рис. 18 и 19).

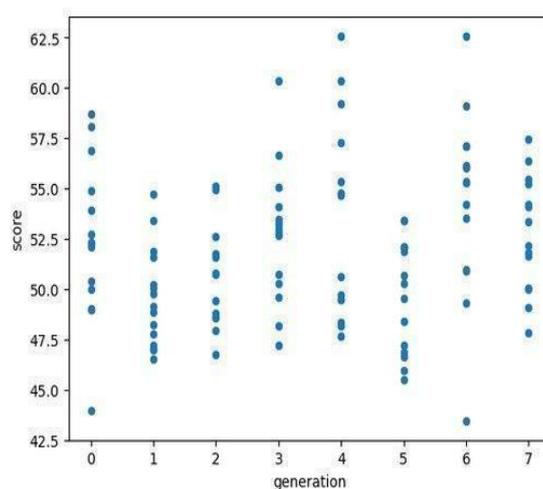


Рис. 18. Сходимость генетического алгоритма по поколениям

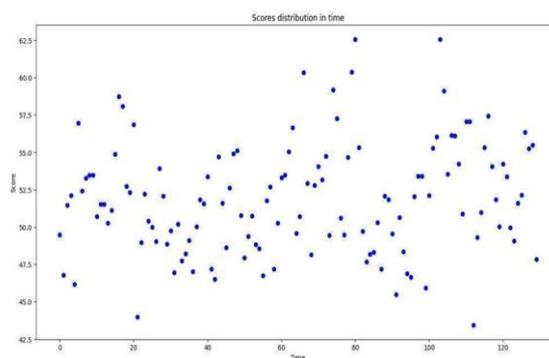


Рис. 19. Сходимость генетического алгоритма по итерациям

Проведенное обучение позволяет выделить решения с высокой степенью оценки, которые используются при построении обучающего сигнала. Анализ полученных значений не показал сходимость параметров как в симуляции, так и на реальном объекте. В симуляции это видно неявно, однако последние поколения наполнены по большей части параметрами с невысокой ошибкой. На реальном объекте явно наблюдается сходимость алгоритма. Для анализа логов и сбора параметров для формирования нечеткого и квантового регуляторов были созданы парсеры. При помощи них формируются .csv файлы, включающие в себя *Time*, *Err*, *errD*, *I*, *kP*, *kI*, *kD* для крена, тангажа и рыскания (табл. 3).

Табл. 3. Пример файла для получения обучающей выборки

	Time	Err	errD	I	kP	kI	kD
1249	91384908	-0.22953	-19.1066	-0.12858	0.423314	0.436126	0.004995
1252	91485153	0.269645	26.49694	-0.12914	0.423314	0.436126	0.004995
1255	91584954	-0.09248	-6.10592	-0.13044	0.423058	0.426715	0.004996
1258	91685140	-0.1404	-14.3225	-0.13155	0.423058	0.426715	0.004996
1261	91784934	0.125885	21.18874	-0.1328	0.423058	0.426715	0.004996
1264	91885120	-0.03555	-1.5238	-0.1352	0.423058	0.426715	0.004996
1267	91984889	-0.0851	-9.11258	-0.13614	0.423058	0.426715	0.004996
1270	92085134	-0.1266	-15.8207	-0.14026	0.423058	0.426715	0.004996
1273	92185005	-0.01785	4.979393	-0.14165	0.423058	0.426715	0.004996
1276	92285115	0.025449	16.70548	-0.14859	0.423058	0.426715	0.004996
1279	92384944	-0.02026	11.97673	-0.15274	0.423058	0.426715	0.004996
1282	92485112	-0.20367	-23.3488	-0.15562	0.423058	0.426715	0.004996
1285	92584896	0.131986	21.33804	-0.1569	0.423058	0.426715	0.004996
1288	92685165	-0.1716	-10.9479	-0.15931	0.423058	0.426715	0.004996

Далее представлены графики обучающих выборок, для обучения нейронных сетей.

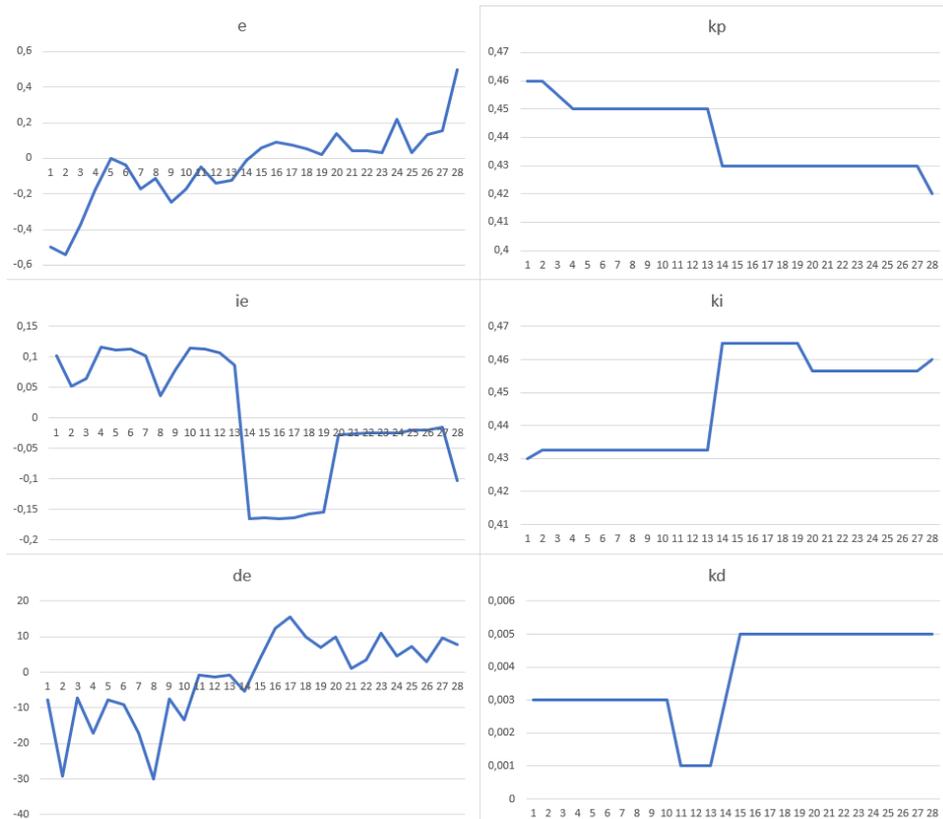


Рис. 20. Сигнал обучения для регулятора тангажа состоящий из ошибки управления, интеграла ошибки, скорости ошибки и соответствующих коэффициентов

На рис. 20 представлен обучающий сигнал, полученный с реального объекта управления в типовой ситуации обучения для регулятора тангажа.

Обучающий сигнал является источником знаний и представляет собой массив данных, разделенный на входные и выходные составляющие, каждая из которых, в свою очередь, состоит из одного и более сигналов. В общем виде, каждый из составляющих сигналов является выборочной (репрезентативной) траекторией некоторого случайного процесса. При этом подразумевается, что в каждый момент времени существует некоторая корреляция между входными и выходными сигналами. Например, в случае аппроксимации некоторого управляющего сигнала, входными компонентами могут являться ошибка управления, интеграл ошибки и ее производная, а выходным компонентом – требуемое значение управляющего воздействия, либо некоторые настраиваемые параметры системы управления, например коэффициенты усиления ПИД-регулятора.

### 3. Проектирование интеллектуальной системы управления на основе мягких и квантовых вычислений

В общем виде структурно ИСУ БВС можно представить в виде следующей схемы (рис. 21):

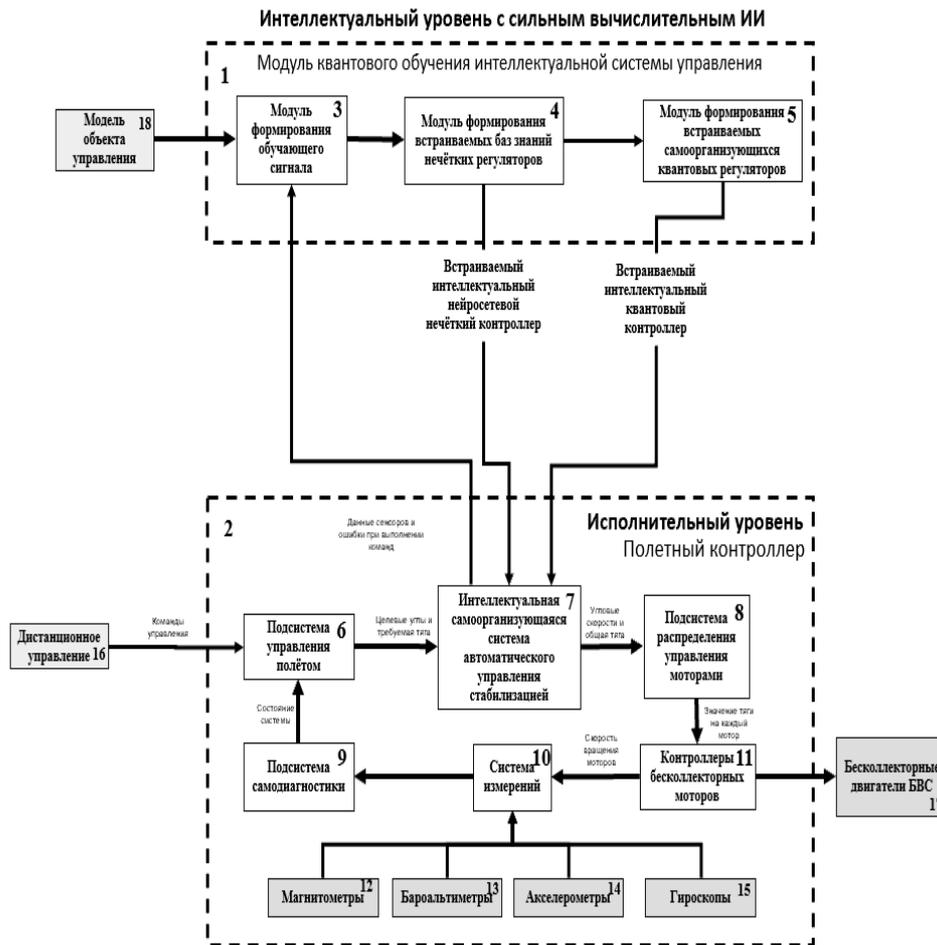


Рис. 21. Схема интеллектуального уровня с ИИ

Программное обеспечение системы управления, в основе которого заложена отечественная сквозная квантовой информационной технологии (КИТ–технология) реализовано на классических процессорах. Такие процессоры используются в типовых БВС, это стало возможно благодаря оптимизации описания квантового алгоритма, представленного в виде простой формулы, состоящей из трех квантовых операторов – суперпозиция, оракул, интерференция, и запрограммированной на языке C++. Как будет представлено далее, применение технологии позволяет обеспечить систему управления БПЛА высоким уровнем устойчивости, управляемости и робастности.

Указанный технический результат достигается тем, что в полетный контроллер БПЛА интегрируется интеллектуальная квантовая самоорганизующаяся система автоматического управления стабилизацией, которая обобщает сигналы управления различных нечётких нейронных контроллеров, спроектированных для разных ситуаций обучения и формирует новый робастный сигнал изменения коэффициентов усиления в реальном времени.

На вход интеллектуальной квантовой самоорганизующейся системы автоматического управления поступают значения ошибок отклонения, интегральной ошибки отклонения и скорости отклонения по крену, тангажу, и рысканью. Далее эти значения поступают в соответствующие интеллектуальные нечеткие регуляторы с заранее спроектированными для разных условий функционирования базами знаний.

При этом, базы знаний нечётких регуляторов для различных ситуаций управления формируются в блоке формирования встраиваемых баз знаний нечётких регуляторов на основе обучающего сигнала, который может быть получен как с помощью модели БВС, так и использованием реальных данных о состоянии системы (из натуральных испытаний).

В процессе функционирования результаты нечёткого логического вывода с нечётких контроллеров поступают в блок самоорганизации неточных баз знаний. Самоорганизация баз знаний

реализуется на основе квантового нечёткого вывода. Выходом из блока квантового вывода являются новые робастные коэффициенты усиления  $PID$ -регуляторов крена, тангажа, рысканья.

В блоке самоорганизации баз знаний выполняется нормирование сигнала по заранее полученным траекториям изменения сигналов для каждого индивидуального нечеткого контроллера. Далее за счет преобразования Адамара и интегральной функции распределения вероятности выполняется формирование квантового бита. Следующим этапом выполняется формирование запутанного (смешанного) состояния, на основе заранее определенного оператора корреляции в виде корреляционной матрицы. Далее применяется оператор оракула – выполняется подготовка амплитуд вероятности состояний, далее в блоке выбора состояния выбирается состояние с максимальной амплитудой, далее в блоке измерения результатов рассчитываются новые значения коэффициентов усиления регуляторов. Таким образом формируется новые обобщенные робастные значения коэффициентов усиления регуляторов крена, тангажа и рысканья.

### 3.1. Процесс проектирования на мягких вычислениях

Применительно к проектированию ИСУ, мягкие вычисления представляют собой комбинацию из следующих подходов: теория нечетких системы, генетические алгоритмы (ГА) и нечеткие нейронные сети (ННС).

НР является центральным элементом ИСУ и вырабатывает управляющие сигналы изменения во времени (законы управления) коэффициентами усиления  $k_p, k_d, k_i$  ПИД-регулятора. Ядро НР основано на процессе проектирования БЗ, которая интегрирована в интеллектуальный контроллер. Структурно БЗ состоит из продукционных логических правил, которые включают нечеткие входные и выходные значения функции принадлежности лингвистических переменных.

Функциональная структура ИСУ с блоками НР и ОБЗ на мягких вычислениях представлена на Рис. 22.

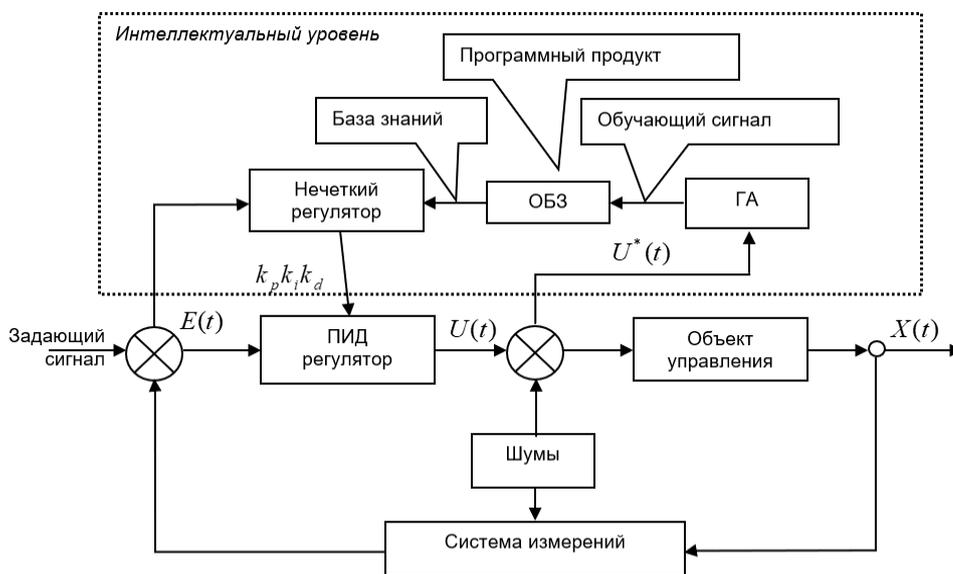


Рис. 22. Структура ИСУ с нечетким регулятором и оптимизатором баз знаний

Такая структура включает одновременно следующие качества управления: управляемость, точность и устойчивость (нижний уровень управления – САУ) и обучение и адаптация (верхний интеллектуальный уровень управления – НР с БЗ).

Наиболее актуальной проблемой при создании ИСУ является получение оптимальной и робастной БЗ, которая гарантирует достижение требуемого уровня качества управления. При проектировании БЗ возникают трудности, связанные со сложностью динамической системы и наличием шумов в системе измерения и управления.

Входом ОБЗ является обучающий сигнал (ОС), который получен на предыдущем этапе экспериментально, т.е. непосредственно из результатов измерений динамических параметров физической модели.

Входными данными для НР является вектор ошибки, который состоит из ошибки управления  $e(t)$ , интеграла ошибки управления  $ie = \int_0^t e dt$  и скорости изменения ошибки управления  $\dot{e}(t)$ . Выходными данными НР является вектор, состоящий из значений коэффициентов усиления  $k_p$ ,  $k_d$ ,  $k_i$  ПИД-регулятора, значения которых применяются в формировании управляющего воздействия в виде:

$$u(t) = k_p(t)e(t) + k_i(t) \int_0^t e(\tau) d\tau + k_d(t)\dot{e}(t). \quad (1)$$

Структурно ОБЗ состоит из генетических алгоритмов (ГА<sub>1</sub>, ГА<sub>2</sub>, ГА<sub>3</sub>), оптимизирующих отдельные компоненты БЗ. Базовые шаги оптимизации и структура ОБЗ представлены на рис. 22. Рассмотрим шаги алгоритма оптимизации [13]:

Шаг 1: Выбор модели нечёткого вывода. Пользователь определяет тип нечёткой модели вывода (Сугено, Мамдани, и т.д.), и число входных и выходных переменных.

Шаг 2: Создание лингвистических переменных. С помощью ГА1 определяется оптимальное число функций принадлежности для каждой входной лингвистической переменной, а также выбирается оптимальная форма представления ее функций принадлежности (треугольная, Гауссовская и т.д.).

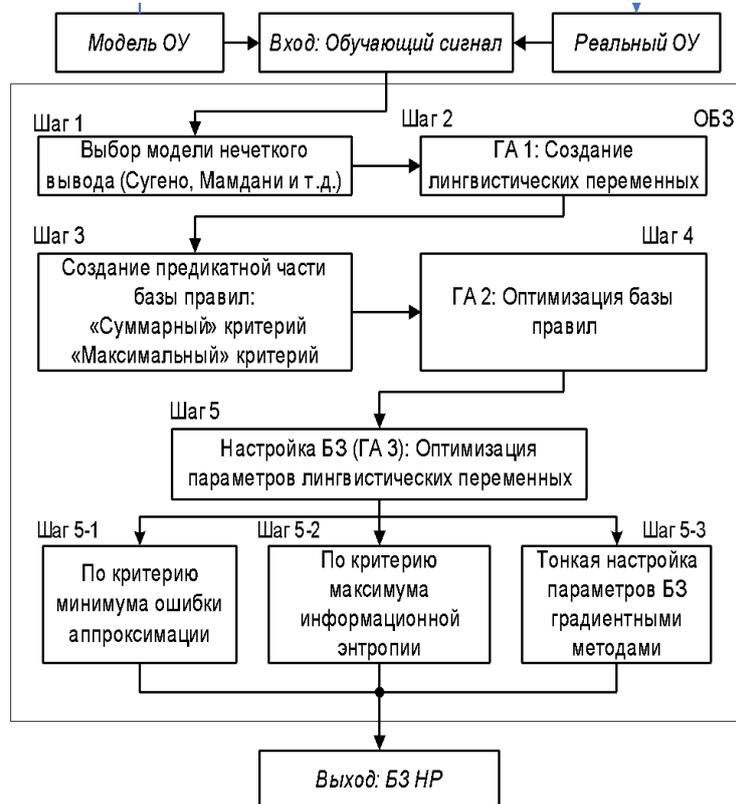


Рис. 23. Структура ОБЗ и шаги оптимизации

Шаг 3: Создание базы правил. На данном этапе используется специальный алгоритм отбора наиболее «робастных правил» в соответствии со следующими двумя критериями:

«суммарный» критерий: выбрать только те правила, которые удовлетворяют следующему условию:

$$R_{total\_fr}^l \geq TL, \quad (2)$$

где  $TL$  (*threshold level*) – заданный (вручную или выбранный автоматически) уровень активации правила, и

$$R_{total\_fr}^l = \sum_{k=1}^N R_{fr}^l(t_k), \text{ и } R_{fr}^l(t_k) = \prod_{j=1}^N [\mu_{j1}^l(x_1(t_k)), \mu_{j2}^l(x_2(t_k)), \dots, \mu_{jn}^l(x_n(t_k))], \quad (3)$$

где  $t_k$  – моменты времени,  $k = 1, \dots, N$ ,  $N$  – равно числу точек в управляющем сигнале;  $\mu_{jk}^l(x_k)$ ,  $k = 1, \dots, N$  – функции принадлежности входных переменных,  $l$  – индекс правила в БЗ; символ «П» обозначает операцию нечеткой конъюнкции (в частности, может интерпретироваться как произведение) «максимальный» критерий: выбрать только те правила, которые удовлетворяют условию:

$$\max_t R_{fr}^l(t) \geq TL. \quad (4)$$

Шаг 4: Оптимизация базы правил. С помощью ГА2 оптимизируются правые части правил БЗ. На данном этапе находится решение, близкое к глобальному оптимуму (минимум ошибки аппроксимации ОС). С помощью следующего шага 5 это решение может быть локально улучшено. В данной работе рассматривается возможность подключения реального удаленного ОУ для оптимизации БЗ в режиме реального времени.

Шаг 5: Настройка базы правил. С помощью ГА3 оптимизируются левые и правые части правил БЗ, т.е. подбираются оптимальные параметры функций принадлежности входных/выходных переменных (с точки зрения заданной функции пригодности ГА). В данном процессе оптимизации используются различные функции пригодности, выбранные пользователем (шаги 5–1 и 5–2 на рис. 23). На данном этапе имеется возможность настройки БЗ с помощью традиционного метода обратного распространения ошибки (см. шаг 5–3 на рис. 23).

### 3.2. Пример построения нечеткого полетного контроллера БВС

Результатом применения оптимизатора баз знаний на мягких вычислениях – *SCOptKB<sup>TM</sup>* – является программный код, который реализует структуру базы знаний нечеткого регулятора.

На рис. 24 – 28 представлены результаты работы ОБЗ на мягких вычислениях для полетного контроллера.

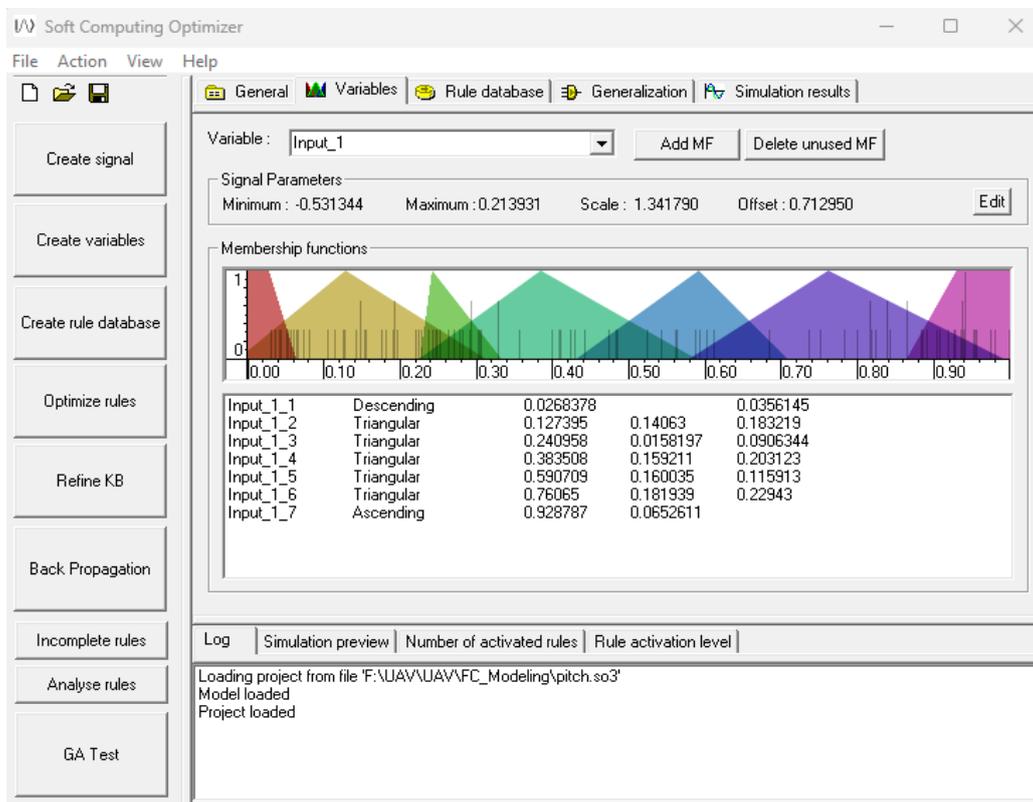


Рис. 24. Интерфейс оптимизатора на мягких вычислениях

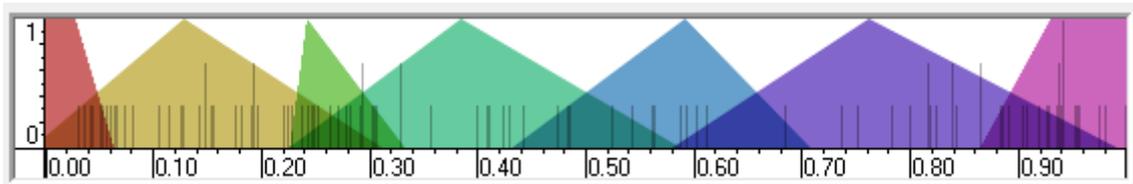


Рис. 25. Нечёткие множества входных переменных по углу крена.

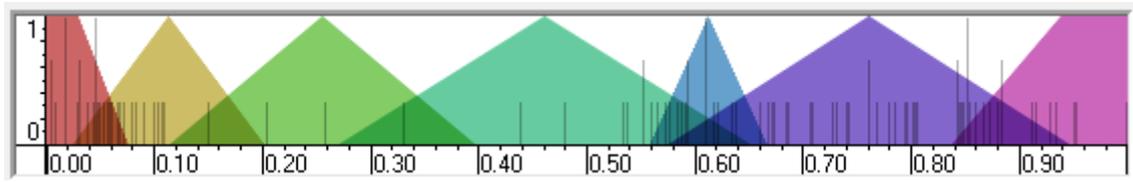


Рис. 26. Нечёткие множества входных переменных по углу тангажа.

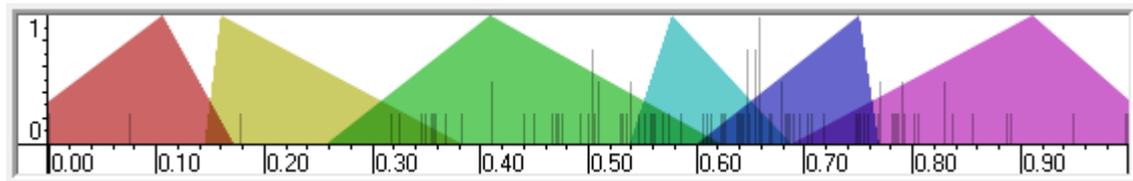


Рис. 27. Нечёткие множества входных переменных по углу рыскания

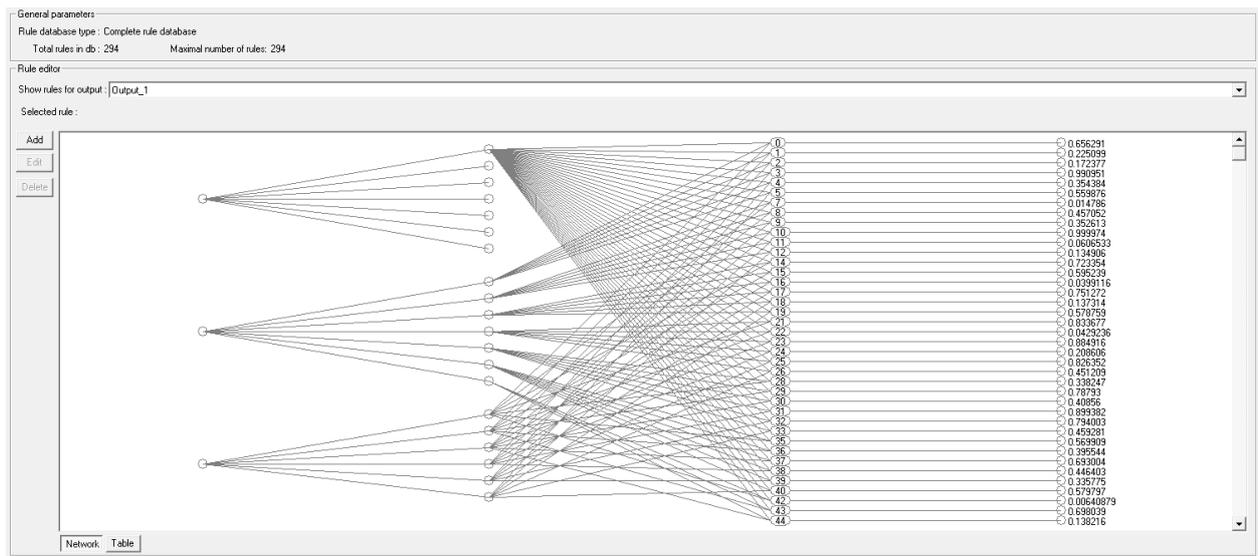


Рис. 28. Структура нечёткой нейронной сети

Данный код интегрируется в полетный контроллер и поддерживается разработанными библиотеками интеллектуального управления БВС.

#### 4. Процесс проектирования квантового регулятора на квантовых мягких вычислениях

Структурно реализованный на классическом процессоре в общем виде квантово-подобный алгоритм (КПА) базируется на основных квантовых операторах теории квантовых вычислений: суперпозиции классических состояний; операторе формирования запутанных состояний (или квантовый оракул); интерференции, а также (классических необратимых) измерениях. Процесс проектирования робастных БЗ соответствует упомянутому выше процессу оптимальной поддержки введенного термодинамического соотношения между отмеченными фундаментальными качествами управления (*thermodynamics control quality trade-off*, рассматриваемый как физический критерий самоорганизации).

Рассмотрим модифицированную модель КНВ (см. рис. 29) как частный случай КА управления самоорганизацией знаний.



Рис. 29. Структурная реализация процесса самоорганизации в модели КНВ: функциональная структура модели КНВ со встроенным Гровер – подобным алгоритмом

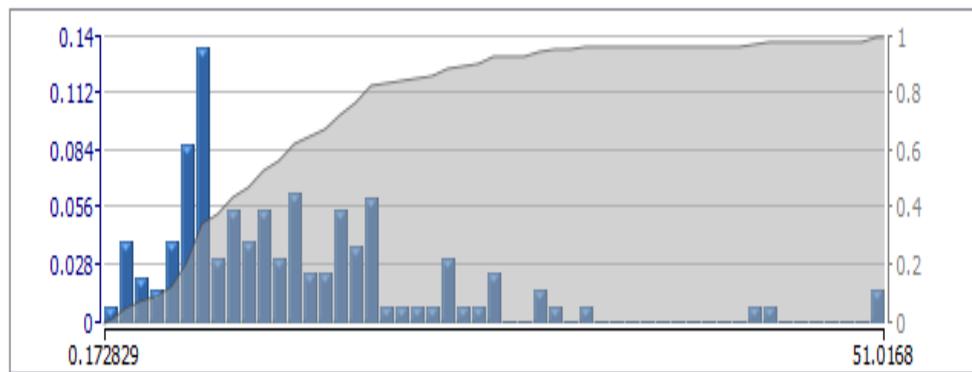
Эффект самоорганизации БЗ в ИСУ основан на виртуальном процессе извлечения дополнительной (скрытой) квантовой информации из реакции (на непредвиденную ситуацию) классических сигналов управления на выходе БЗ, спроектированных в условиях обучения, и физически реализуется программным инструментарием. Элементы самоорганизации показаны в блоках КНВ на. Программно–аппаратная поддержка алгоритма квантовой самоорганизации реализуется за счет инструментария  $QCOptKB^{TM}$ . Для достижения самоорганизации в структуре ИСУ используется КНВ.

На рис. 26 показана функциональная структура модифицированной (на основе Гровер–подобного поискового алгоритма оптимальной корреляции) модели КНВ, состоящая из следующих этапов:

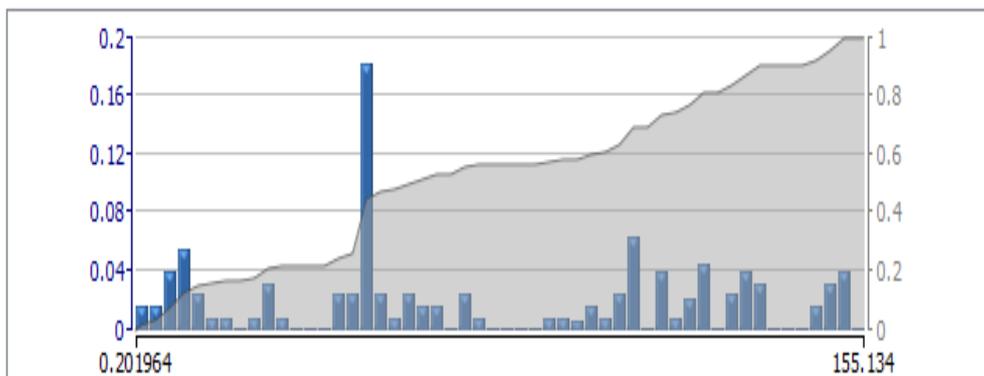
1. обработка результатов нечеткого вывода каждого независимого индивидуального НР;
2. выбор тип квантовой корреляции на основе Гровер – подобного квантового генетического алгоритма;
3. формируется суперпозиция для выбранной квантовой корреляции;
4. извлечение информация, скрытой в независимых индивидуальных БЗ (по принципу минимума энтропии в извлеченных квантовых знаниях – максимальной амплитуды вероятностей «интеллектуального состояния») на основе методов квантовой теории информации;
5. в режиме реального времени проектируется обобщенный выходной робастный сигнал управления на множестве БЗ в виде реакций НР на новую ошибку управления.

В этом случае выходной сигнал КНВ в режиме реального времени представляет оптимальный сигнал управления изменением коэффициентов усиления нечеткого ПИД–регулятора, который включает в себя необходимые (наилучшие) качественные характеристики выходных сигналов управления каждого из НР, реализуя тем самым принцип самоорганизации. Качественные особенности синергетического эффекта самоорганизации учитываются в выбранном типе и виде квантовой корреляции и обозначены на рис. 26 в блоке «Самоорганизация».

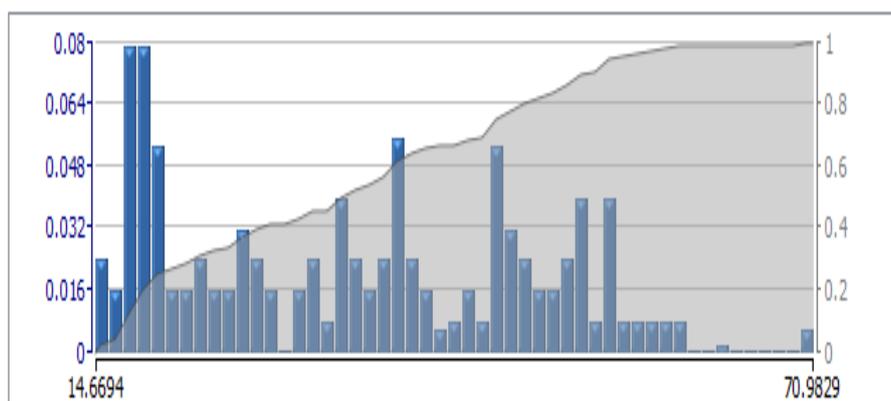
На рис. 30 в качестве иллюстрации представлены гистограммы распределения выходных сигналов нечеткого регулятора. Нормированное интегральное распределение этого сигнала рассматривается как источник информации о состоянии работы НР и используется при формировании логического кубита.



*Пропорциональный коэффициент усиления*



*Интегральный коэффициент усиления*



*Дифференциальный коэффициент усиления*

*Рис. 30. Гистограммы распределения выходных сигналов коэффициентов усиления нечеткого регулятора*

Физически классическая корреляция отвечает за самоорганизацию структуры на макроуровне; квантовая и смешанная корреляции отвечают за микроуровень и передачу информации с микро – на макроуровни соответственно. С помощью различных типов квантовых корреляций (внутренней, внешней, пространственной, временной и пространственно – временной) осуществляется обмен информацией и координационное управление между коэффициентами усиления проектируемого робастного интеллектуального квантового регулятора (см. [14]).

На рис. 31 представлен интерфейс программного инструментария с выбранной пространственной корреляцией.

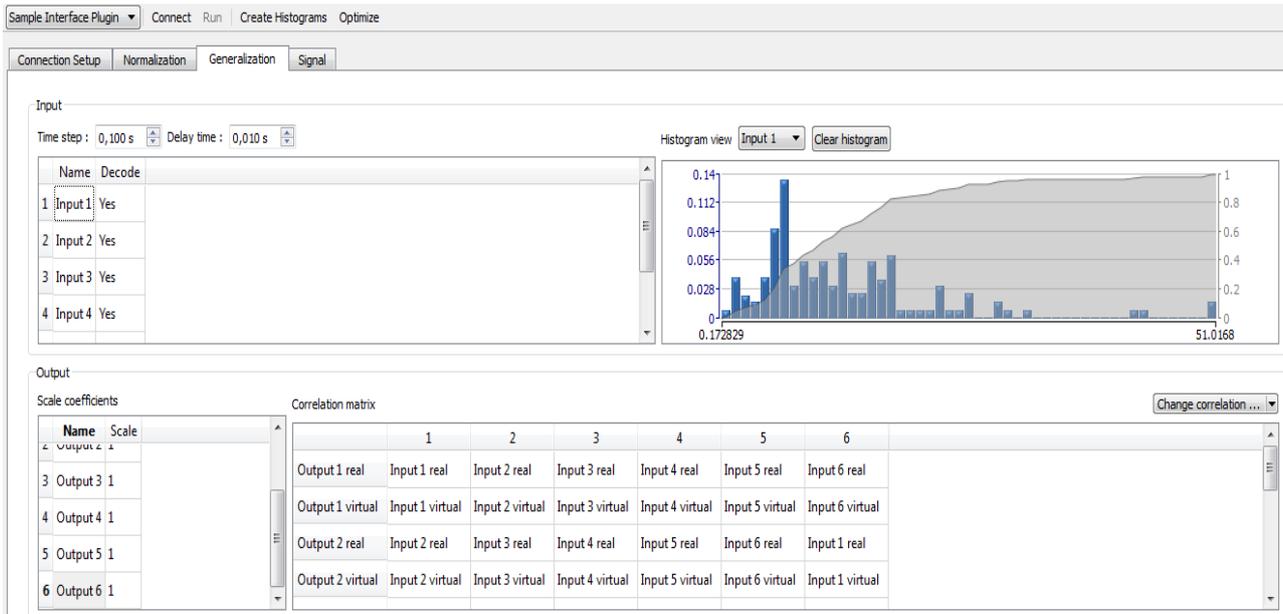


Рис. 31. Интерфейс программного инструментария QCOCK™

Перейдем к практическому применению разработанной модели КНВ для формирования процессов управления коэффициентами усиления ПИД–регуляторов регулировки крена, тангажа и рысканья.

Основная идея применения квантового координационного управления заключается в поиске и использование скрытых взаимосвязей на основе корреляций между работой регуляторов регулировки крена, тангажа и рысканья. Для этих целей, на основе различных ситуаций управления и с применением генетических алгоритмов были спроектированы нечеткие регуляторы [16,17], изменяющие коэффициенты усиления классического ПИД регулятора (см. п. 4).

Далее представлена структура системы управления БВС (рис. 32):

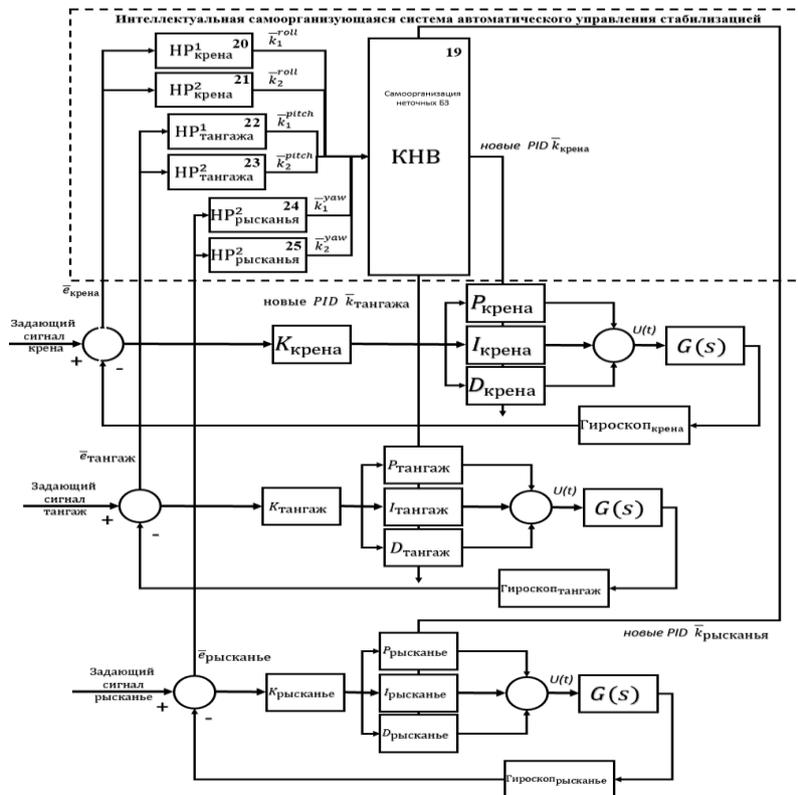


Рис. 32. Структура системы управления БВС

Гистограммы распределения входных сигналов квантового регулятора БВС приведены на рис. 33 – 38.

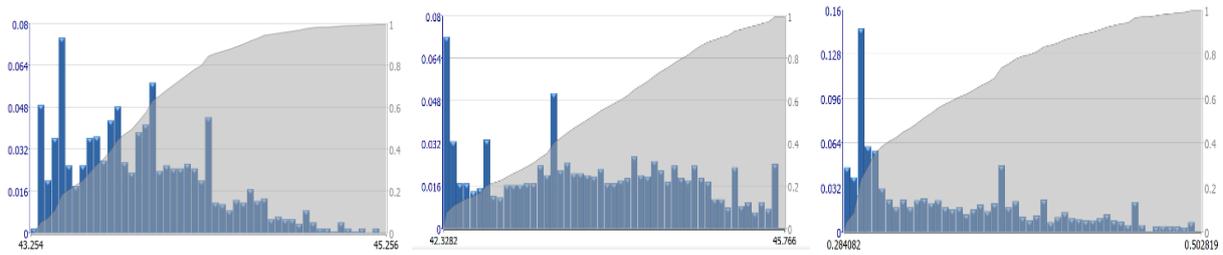


Рис. 33. Гистограммы распределения входных сигналов квантового регулятора БВС

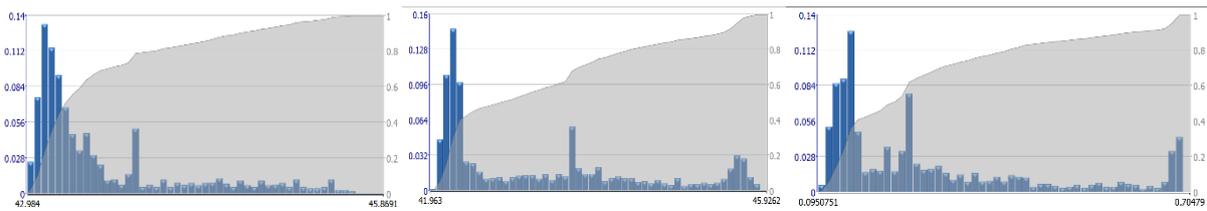


Рис. 34. Гистограммы распределения входных сигналов квантового регулятора БВС

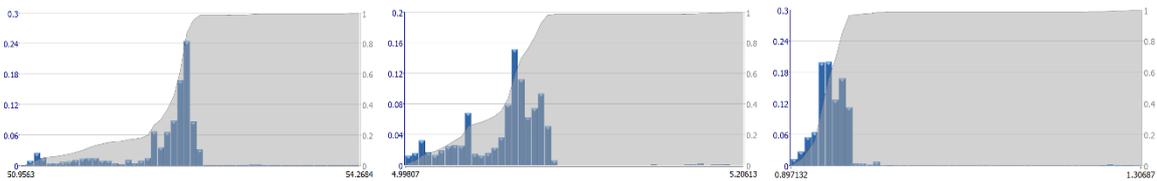


Рис. 35. Гистограммы распределения входных сигналов квантового регулятора БВС

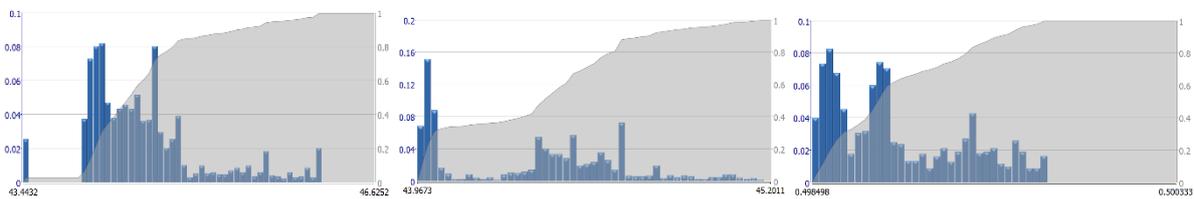


Рис. 36. Гистограммы распределения входных сигналов квантового регулятора БВС

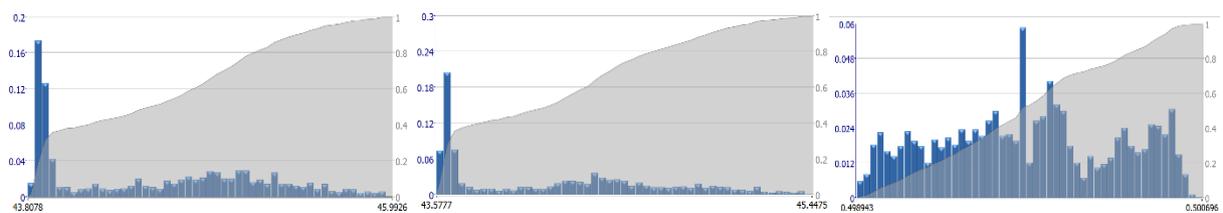


Рис. 37. Гистограммы распределения входных сигналов квантового регулятора БВС

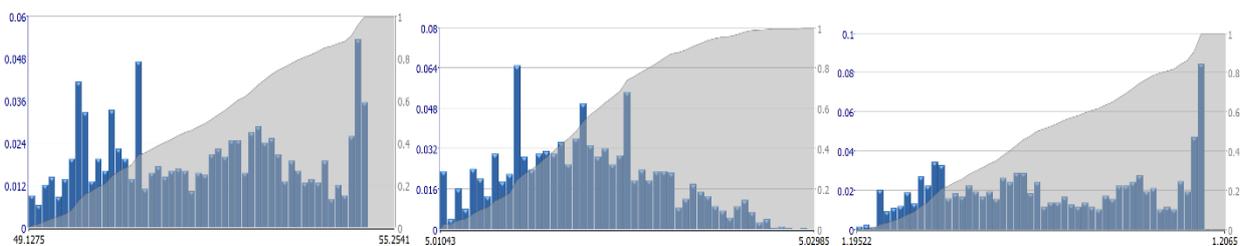


Рис. 38. Гистограммы распределения входных сигналов квантового регулятора БВС

Окно программного инструментария для проектирования квантового регулятора БВС.

На рис. 39 приведен вид окна программного инструментария.

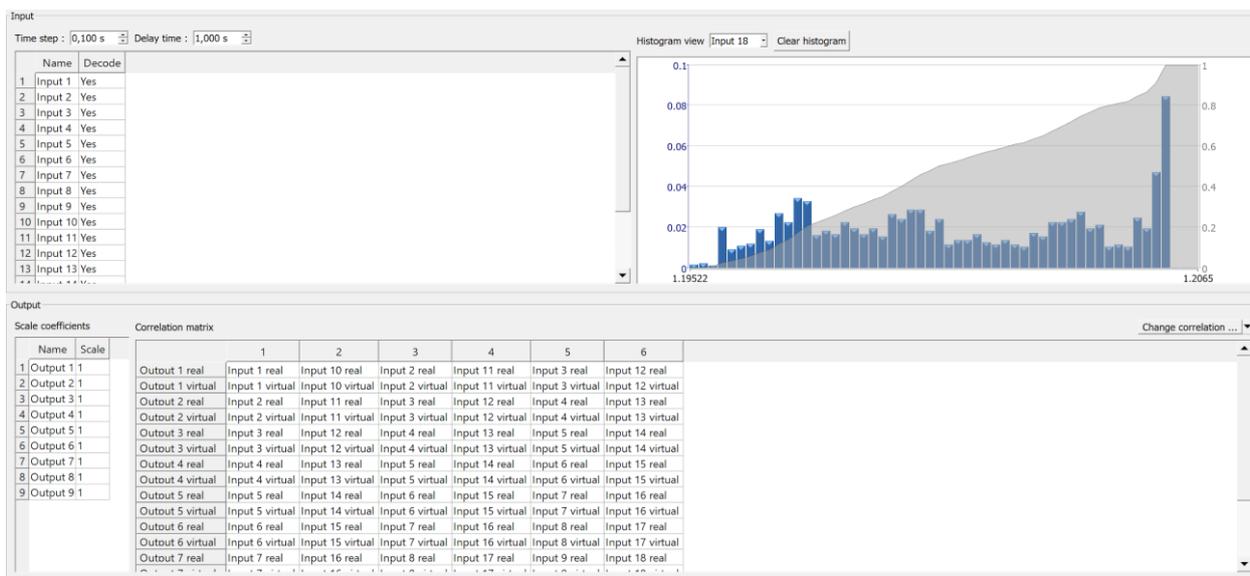


Рис. 39. Окно программного инструментария

В следующем разделе приведены результаты применения рассмотренных систем управления.

## 5. Результаты проведения экспериментов

Исследования качеств управления регуляторов проводилось согласно заданному маршруту с изменениями высоты. В качестве проверяемых регуляторов рассматривались: ПИД, Нечёткие регуляторы с различными базами знаний и квантовые регуляторы. Каждый полёт включает 3 цикла повторения маршрута, на первой и последней маршрутной точке выполняется задержка в висении на 10 секунд, для упрощения последующего сравнительного анализа (рис. 40).

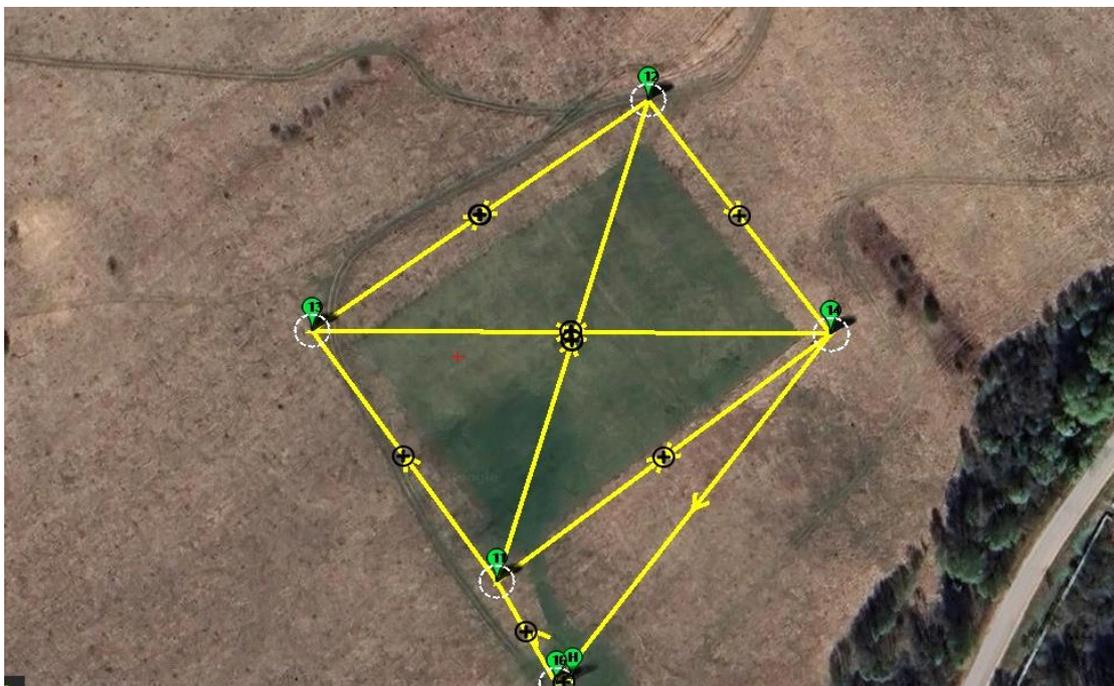


Рис. 40. Маршрут проведения экспериментальных полетов

Исследования проводились в несколько дней с разными условиями по ветру и с аккумуляторной батареей – АКБ – на 8 А/ч, для набора статистики и анализа.

Полет разработан с учетом различных «неудобных» ситуаций для БВС, таких как:

1. Остроугольные повороты
2. Высокая точность прохода поворотов (2 метра)
3. Подъем на 10 метров после стадии 11
4. Спуск на 10 метров после стадии 13

Рассмотрим результаты экспериментов в первой (типовой) ситуации управления С1. Скорость ветра составляла 1–2 м/с. Вес БВС типовой, стандартный.

### 5.1. ПИД регулятор – ситуация С1

На рис. 41 приведены результаты управления БВС на основе ПИД – регулятора.

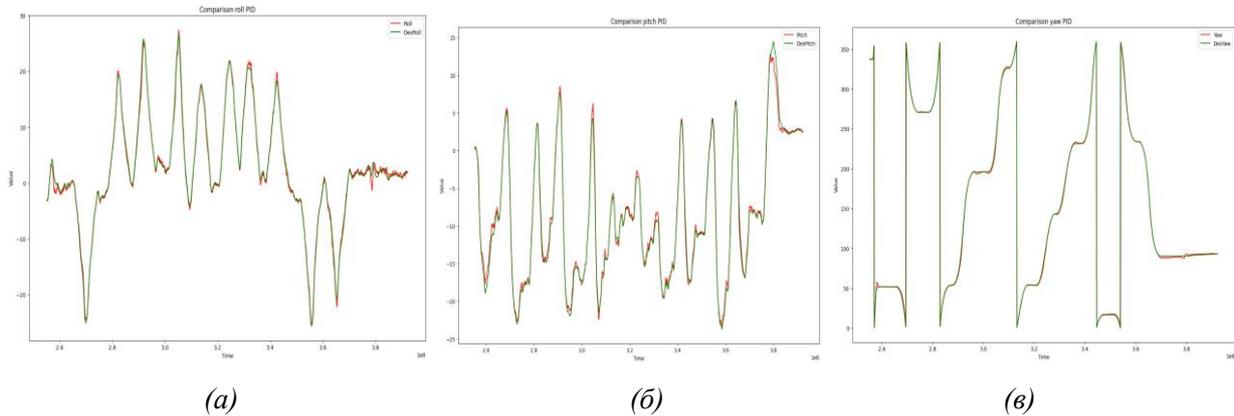


Рис. 41. Сравнение целевого и реального крена (а), тангажа (б) и рысканья (в) на основе ПИД – регулятора

Коэффициенты усиления регуляторов при этом были постоянные (рис. 42)

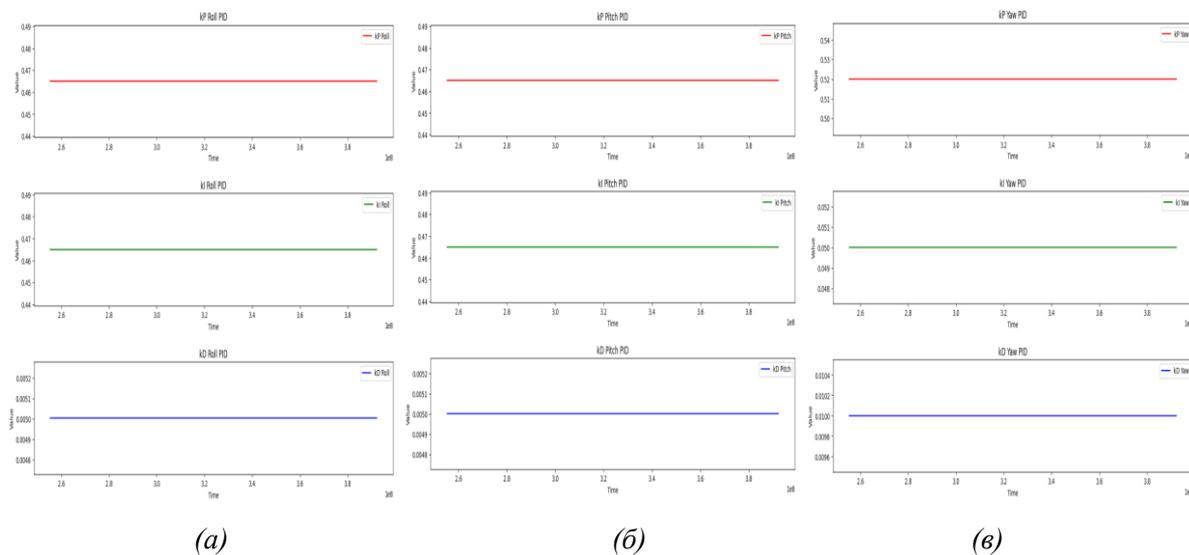


Рис. 42. Коэффициенты усиления ПИД – регулятора (крен (а), тангаж (б), рыскание (в))

На рис. 43 результаты формирования ПИД – регулятором сил управления в виде управляющих команд работой двигателей.

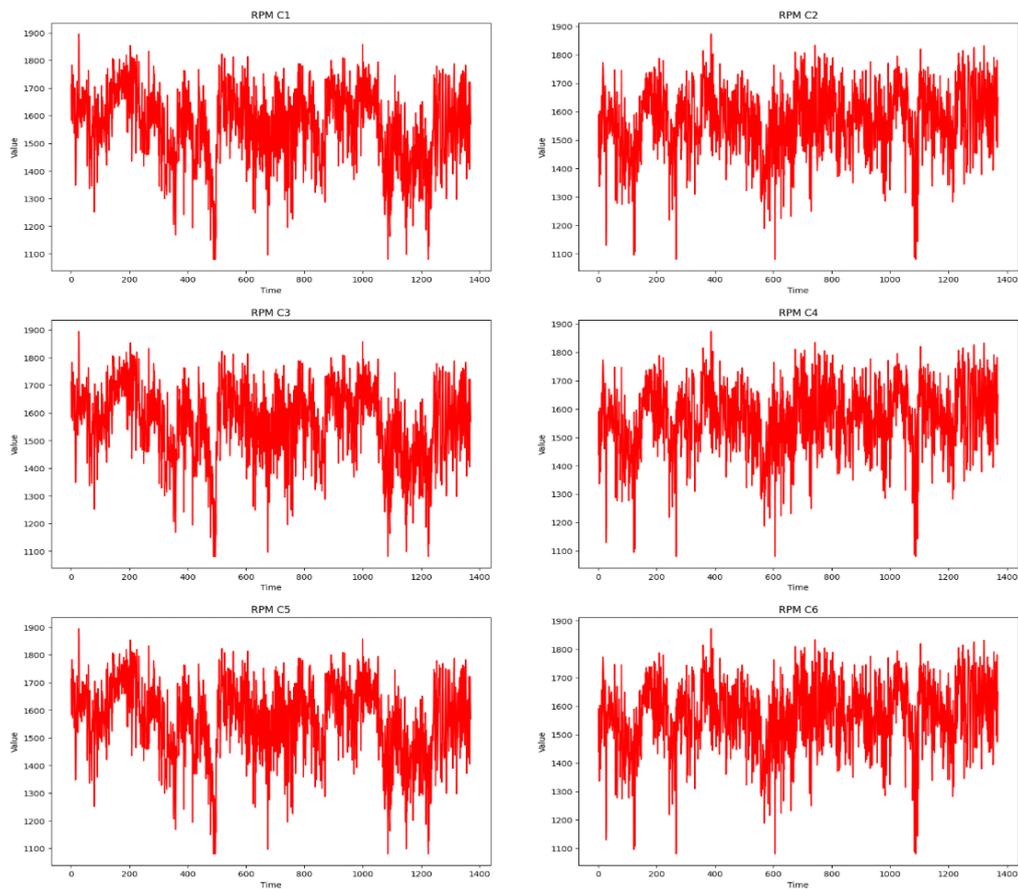


Рис. 43. Управляющие команды двигателям ПИД – регулятором

Рассмотрим поведение БВС при управлении нечетким регулятором в ситуации управления С1.

## 5.2. НЕЧЁТКИЙ КОНТРОЛЛЕР 1 – С1

Поведение БВС при управлении нечётким контроллером для типовой ситуации приведено на рис. 44.

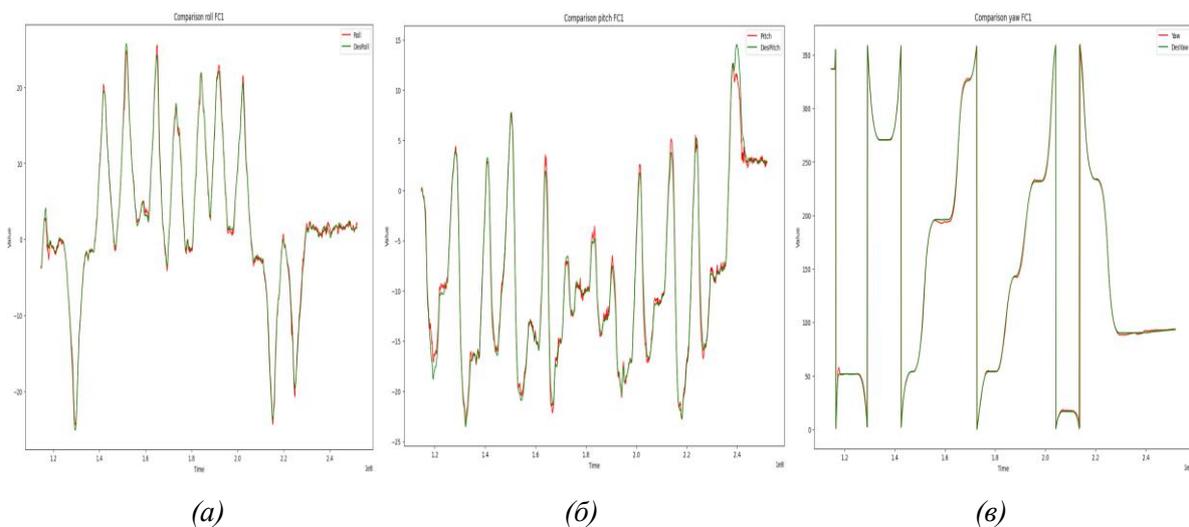
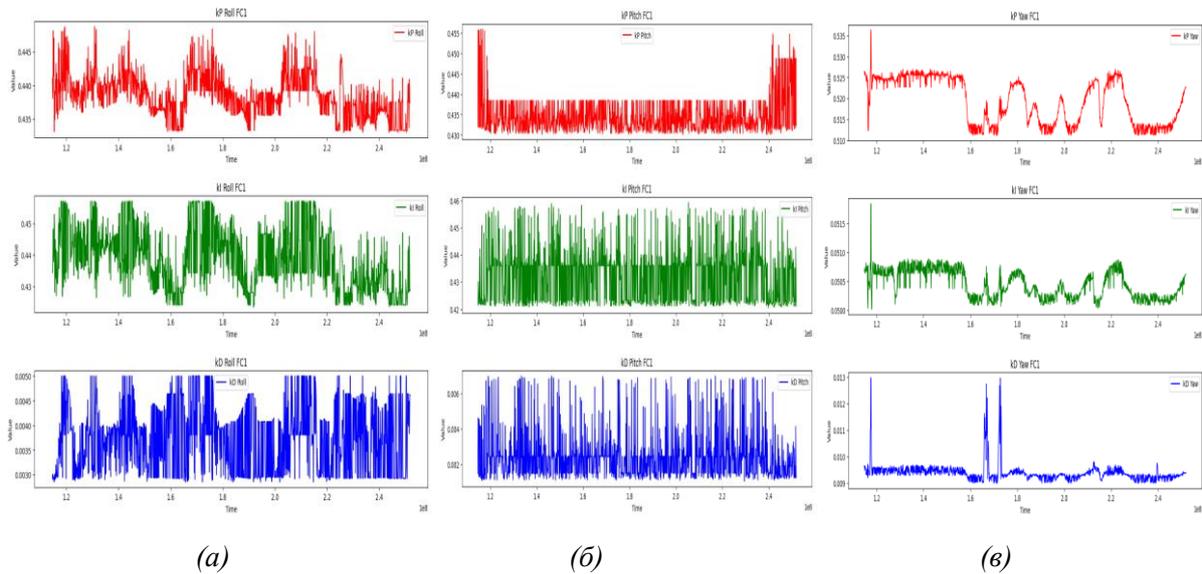


Рис. 44. Сравнение целевого и реального крена (а), тангажа (б) и рыскания при управлении FC1.

Рисунок 45 демонстрирует поведение во времени коэффициентов усиления FC1 (крен (а), тангаж (б), рыскание (в))



(а) (б) (в)  
 Рис. 45. График коэффициентов усиления FC1 (крен (а), тангаж (б), рыскание (в))

Управляющие команды двигателями НР1 приведены на рис. 46.

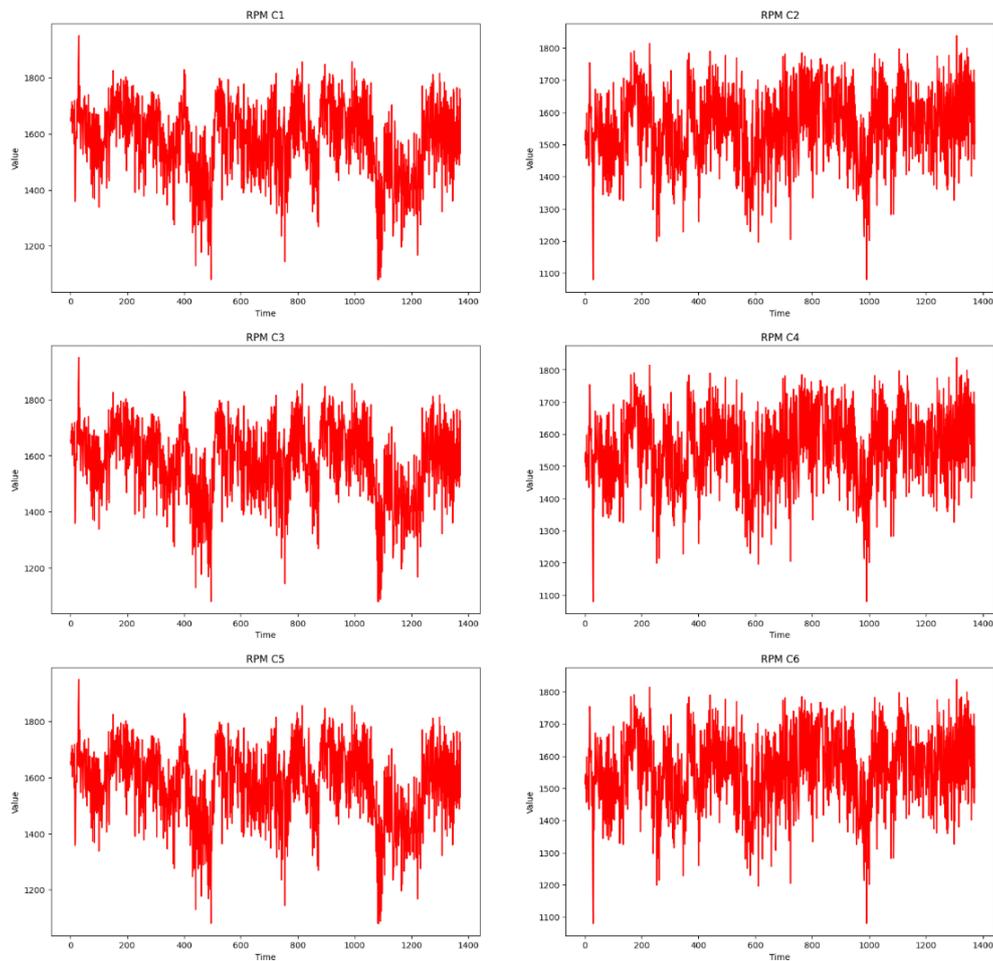


Рис. 46. Управляющие команды двигателями НР1

Рассмотрим поведение БВС при управлении вторым нечетким регулятором в ситуации управления С1.

### 5.3. НЕЧЁТКИЙ КОНТРОЛЛЕР 2 – С1

Нечёткий контроллер НР2 разработан для полета с полезной нагрузкой.

На рис. 47 приведены результаты сравнения целевого и реального значений крена, тангажа и рыскания при управлении БВС с нечетки регулятором FC2.

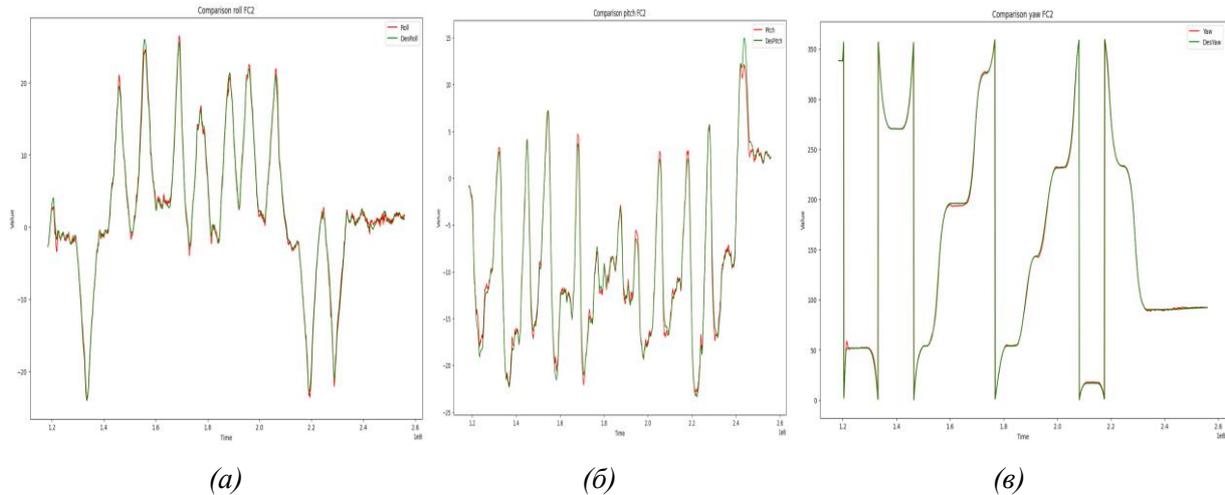


Рис. 47. Сравнение целевого и реального сигналов крена (а), тангажа (б) и рыскания (в) при управлении FC2

Рисунок 48 демонстрирует результаты формирования коэффициентов усиления нечетким регулятором FC2 (крен, тангаж и рыскание, соответственно)

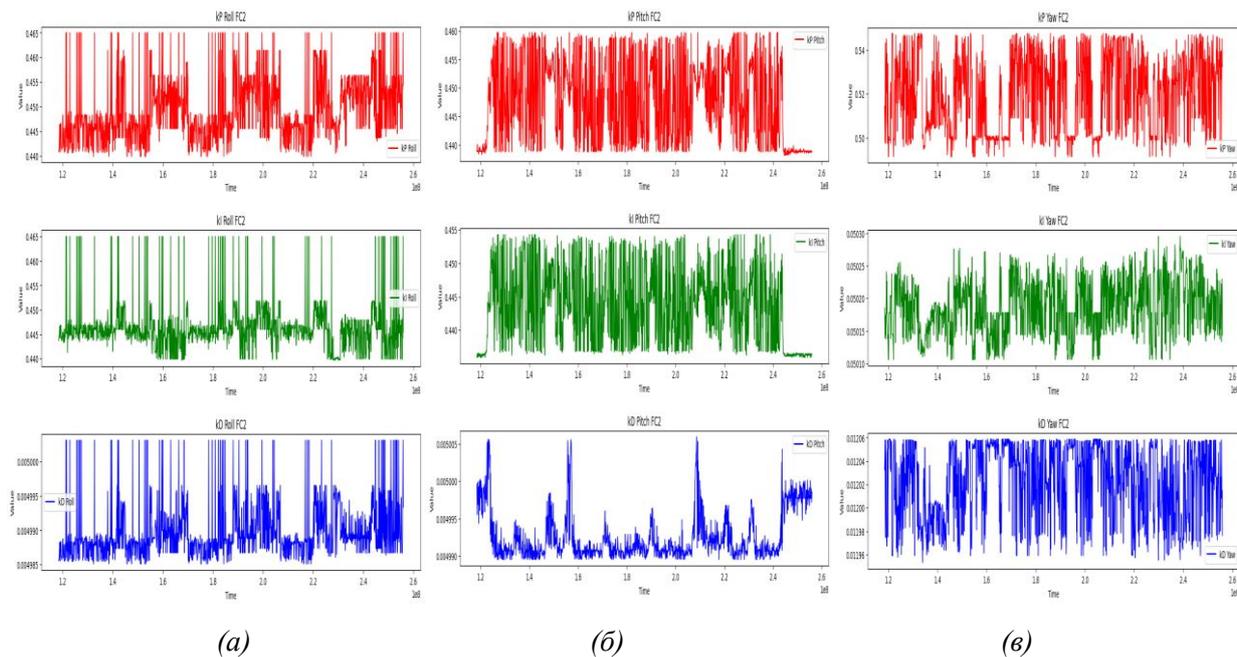


Рис. 48. График коэффициентов усиления при применении FC2 (крен (а), тангаж (б), рыскание (в))

На рис. 49 приведены результаты формирования сил управления двигателями при управлении нечетким регулятором НР2.

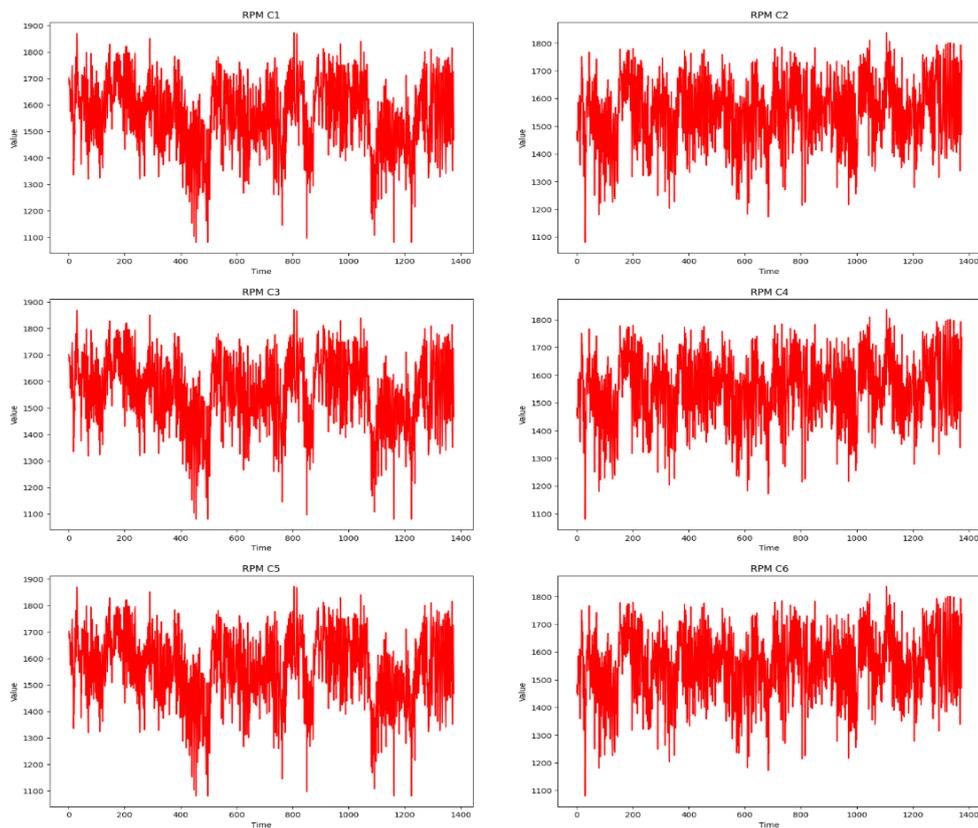
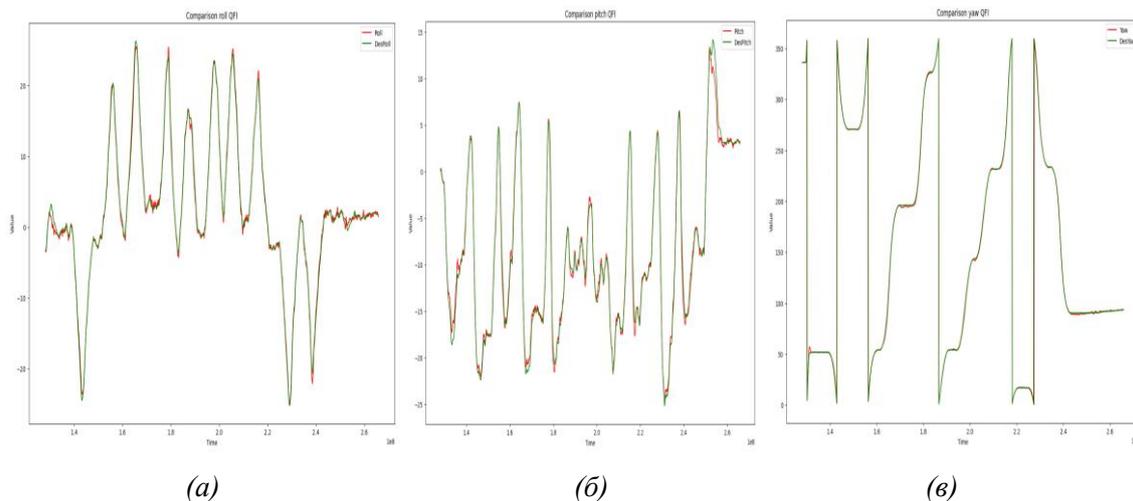


Рис. 49. Управляющие команды двигателям HP2

Рассмотрим далее применение квантового регулятора на основе разработанных нечетких регуляторов  $FC1$  и  $FC2$  в ситуации  $C1$ .

#### 5.4. КВАНТОВЫЙ РЕГУЛЯТОР В $C1$

Результаты сравнения целевого и реального сигналов крена (а), тангажа (б) и рыскания (в) при применении квантового регулятора на основе квантового вывода  $QFI$  приведены на рис. 50.



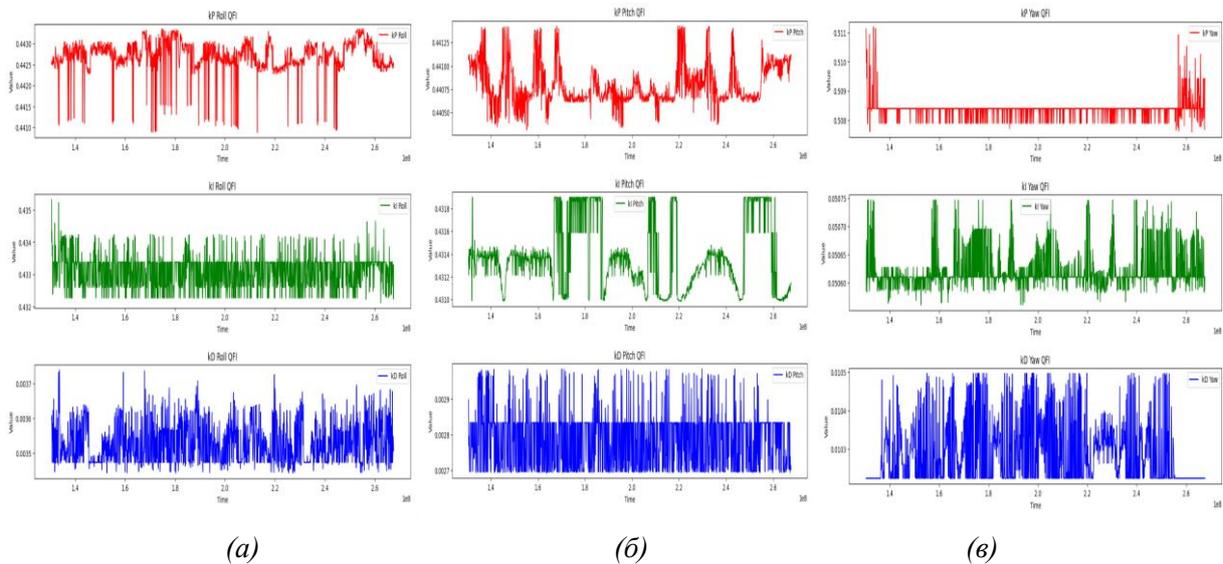
(а)

(б)

(в)

Рис. 50. Сравнение целевого и реального крена (а), тангажа (б) и рыскания (в) при применении модели квантового регулятора на основе  $QFI$

Рисунок 51 демонстрирует поведение коэффициентов усиления ПИД – регулятора при применении  $QFI$  (крен (а), тангаж (б) и рыскание (в))



(а) (б) (в)  
 Рис. 51. График коэффициентов усиления ПИД – регулятора при применении QFI (крен (а), тангаж (б) и рыскание (в))

На рис. 52 приведены результаты формирования управляющих команд двигателями при применение квантового нечеткого вывода QFI.

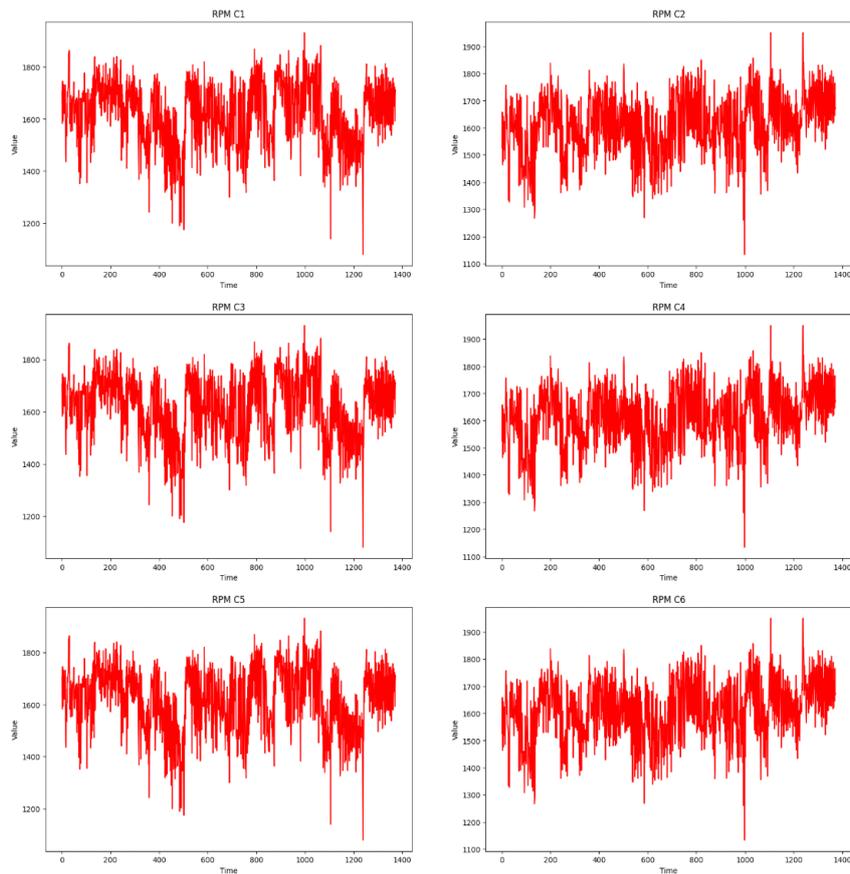


Рис. 52. Управляющие команды двигателями QFI

Рассмотрим прямые и косвенные показатели качества систем управления при различных типах регуляторов.

На рис. 53 приведены результаты сравнения интеграла ошибки регуляторов по высоте при различных типах регуляторов.

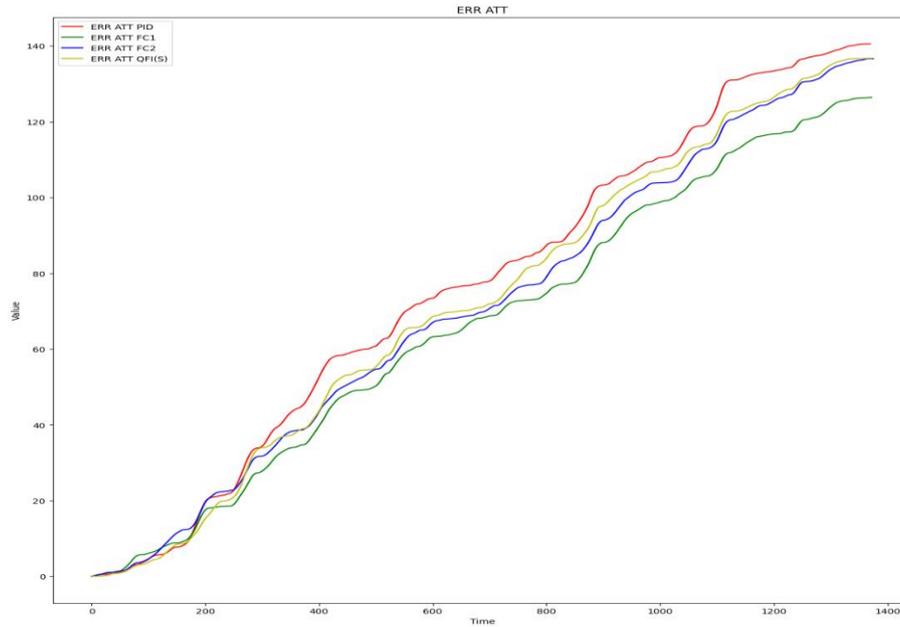


Рис. 53. Интеграл ошибки регуляторов по высоте при различных типах регуляторов

Рисунок 53 показывает, что квантовый регулятор, как и нечёткие контроллеры лучше держат уровень высоты. Лучше всего в типовой ситуации C1 высоту держит нечеткий регулятор НР1.

Интеграл квадратичной ошибки крена приведен на рис. 54.

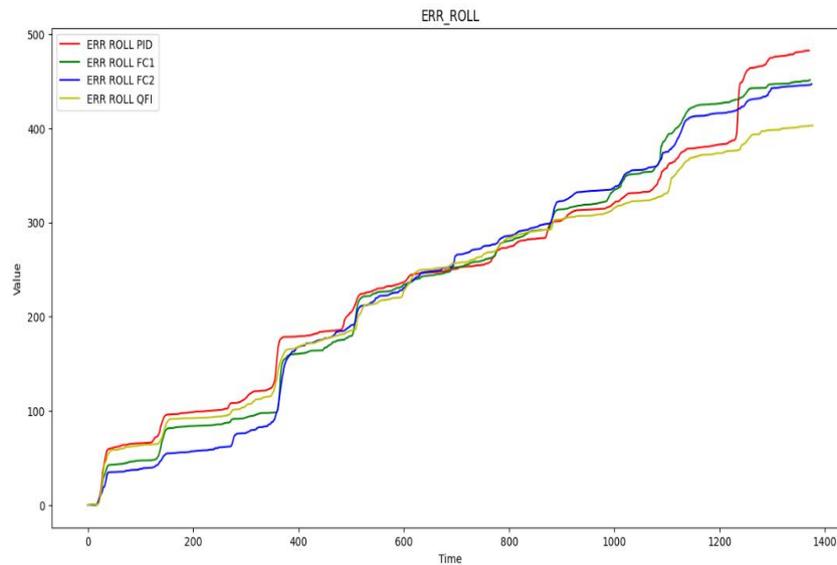


Рис. 54. Сравнение поведения интеграла квадратичной ошибки крена при применении различных типов регуляторов в ситуации C1

Из рис. 54 видно, что квантовый регулятор имеет наименьшую квадратичную интегральную ошибку по крену. Нечеткие регуляторы показывают меньшую ошибку, чем ПИД – регулятор.

На рис. 55 приведен результат накопления ошибки управления в виде интеграла квадратичной ошибки тангажа при различных типах регуляторов.

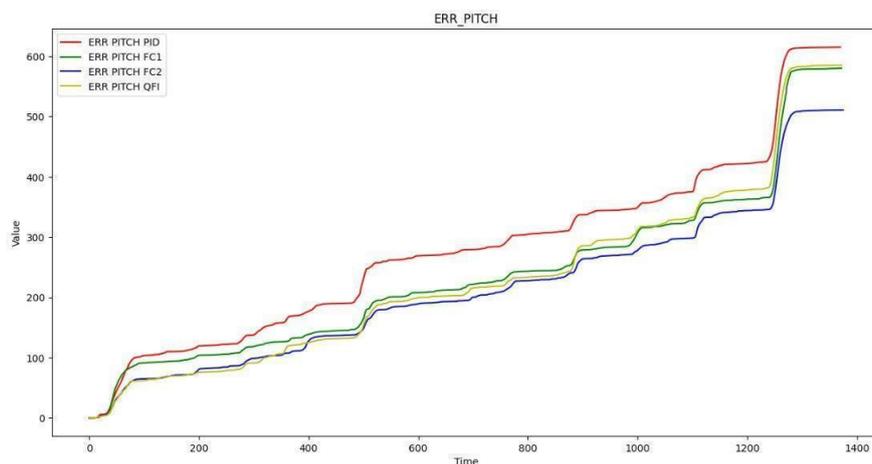


Рис. 55. Интеграл квадратичной ошибки тангажа

Из рис. 55 видно, что один из нечетких регуляторов имеет наименьшую квадратичную интегральную ошибку по тангажу. Оставшийся нечеткий регулятор и квантовый регулятор имеют ошибку меньше, чем ПИД –регулятор.

На рис. 56 приведен результат накопления ошибки управления в виде интеграла квадратичной ошибки рыскания при различных типах регуляторов.

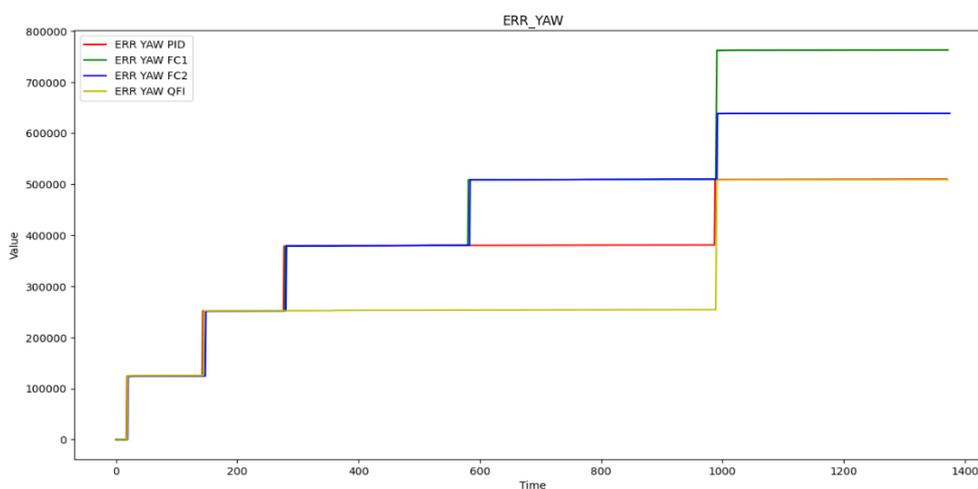


Рис. 56. Интеграл квадратичной ошибки рыскания

Из рис. 56 видно, что квантовый регулятор имеет идентичную ошибку с ПИД – регулятором.

Интегральный показатель качества управления демонстрирует более хорошее управление у квантового регулятора.

Табл. 4. Прямые показатели качества управления

Тип регулятора	Сложность управления	Перерегулирование
ПИД	652	5.63
НР1	613	<b>5.07</b>
НР2	667	5.34
КНВ	<b>565</b>	5.21

Квантовый регулятор, как и нечёткие регуляторы характеризуются более простым управлением, это значит, что команды исполнительным механизмам посылаются более «мягкие». Сложность управления оценивалась по командам передаваемый двигателям.

Теперь перейдем к новой, более сложной непредвиденной ситуации управления *С2*.

Ситуация характеризуется наличием полезной нагрузки (681 грамм) и повышенным ветром 5–7 м/с порывами до 9 м/с.

## 5.5. ПИД регулятор – ситуация С2

Рассмотрим систему управления на ПИД – регуляторе с постоянными коэффициентами.

На рис. 57 приведены результаты сравнения ожидаемого и реального крена в ситуации С2 при применении ПИД – регулятора.

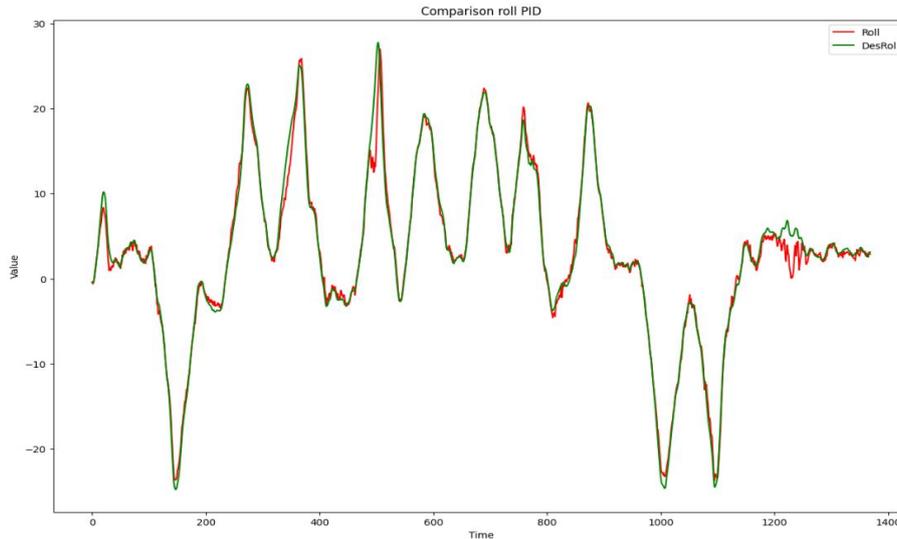


Рис. 57. Сравнение ожидаемого и реального сигнала крена при применении ПИД – регулятора

На рис. 58 приведен результат отклонения угла тангажа при применении ПИД – регулятора.

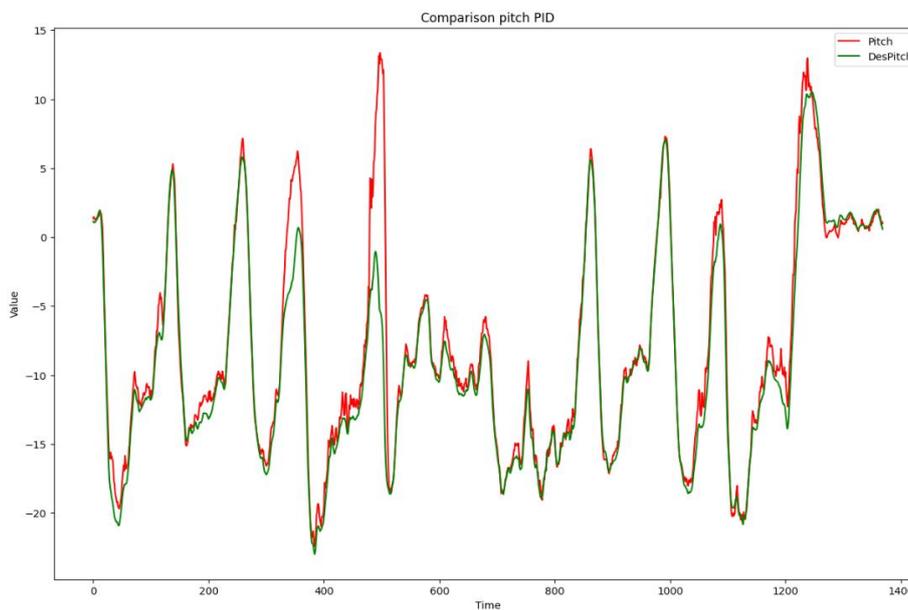


Рис. 58. Сравнение ожидаемого и реального угла тангажа при применении ПИД – регулятора

На рис. 59 приведен результат отклонения угла рыскания при применении ПИД – регулятора.

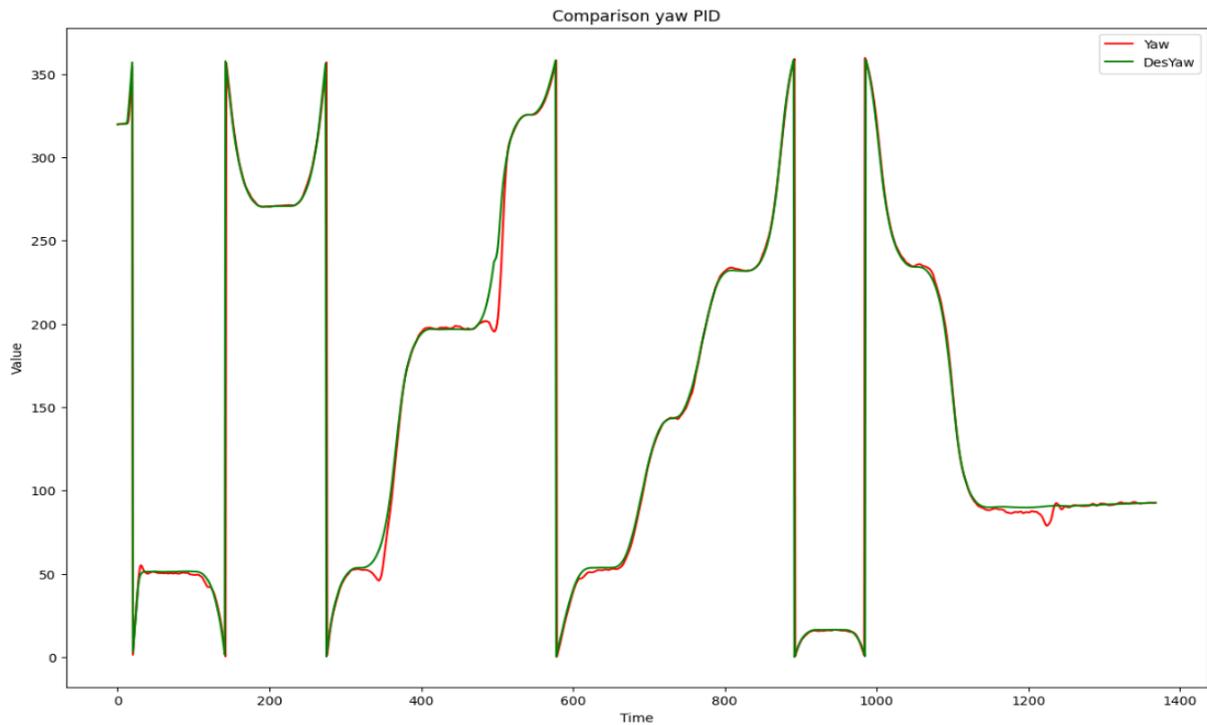


Рис. 59. Сравнение ожидаемого и реального рыскания PID

При этом значения коэффициентов усиления были постоянными (рис. 60).

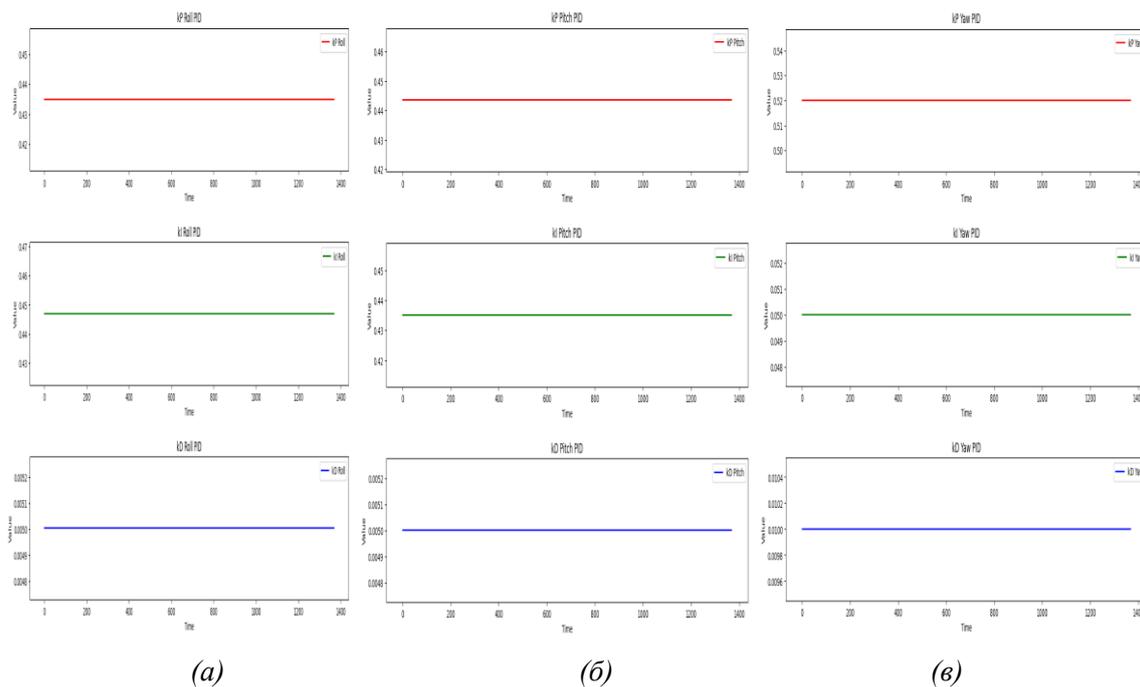


Рис. 60. График коэффициентов усиления ПИД – регулятора (крен (а), тангаж (б) и рыскание (в))

Рассмотрим результаты эксперимента с нечётким регулятором.

### 5.6. Нечеткий регулятор FC1 – C2

На рис. 61 приведен результат поведения угла крена при применении нечеткого регулятора FC1 в ситуации C2.

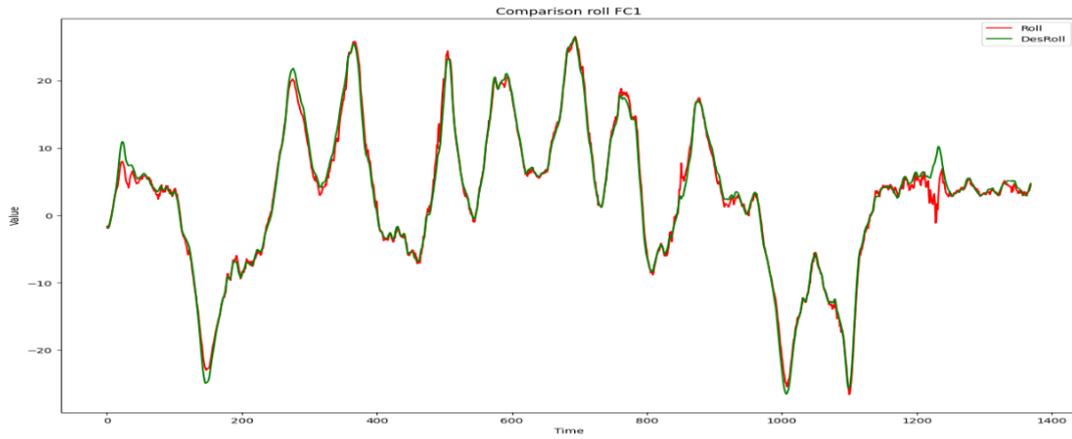


Рис. 61. Сравнение ожидаемого и реального крена FC1

Рисунок 62 демонстрирует результат сравнения ожидаемого и реального угла тангажа для нечеткого регулятора FC1.

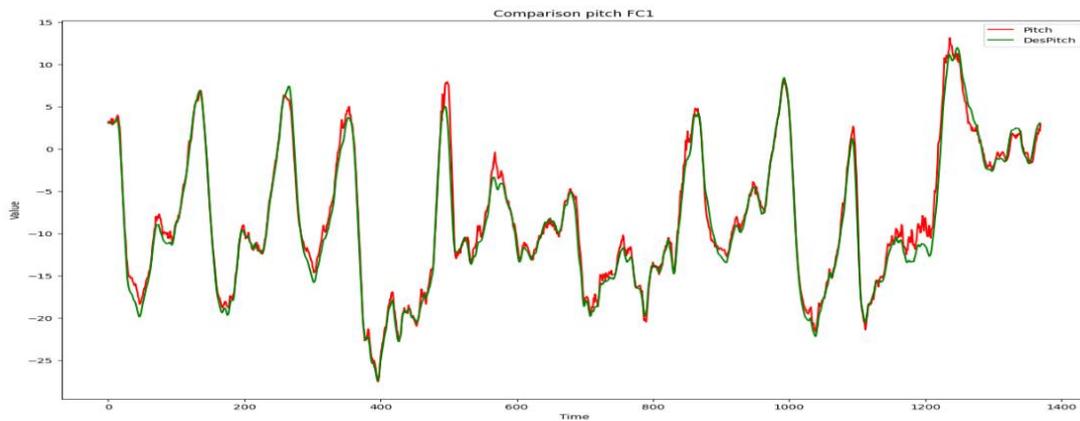


Рис. 62. Сравнение ожидаемого и реального угла тангажа для нечеткого регулятора FC1

Рисунок 63 демонстрирует результат сравнения ожидаемого и реального угла рыскания для нечеткого регулятора FC1.

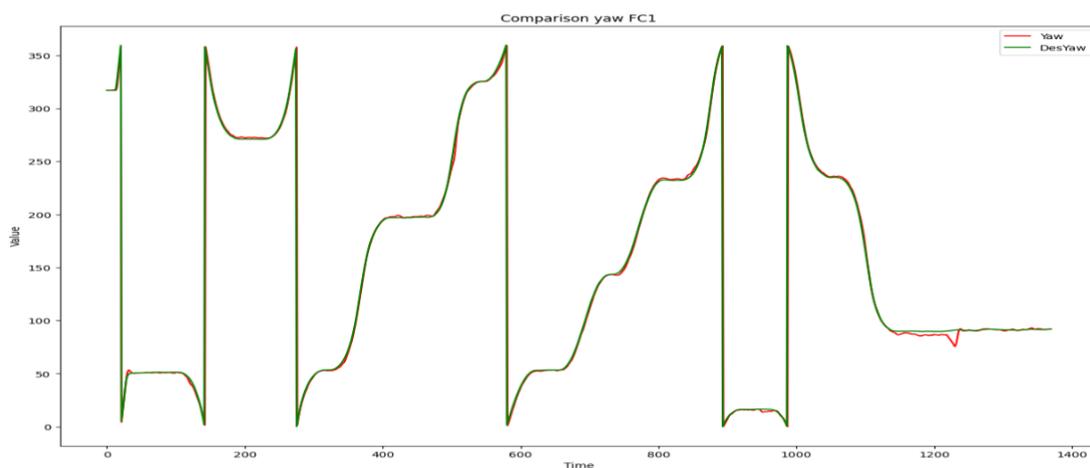
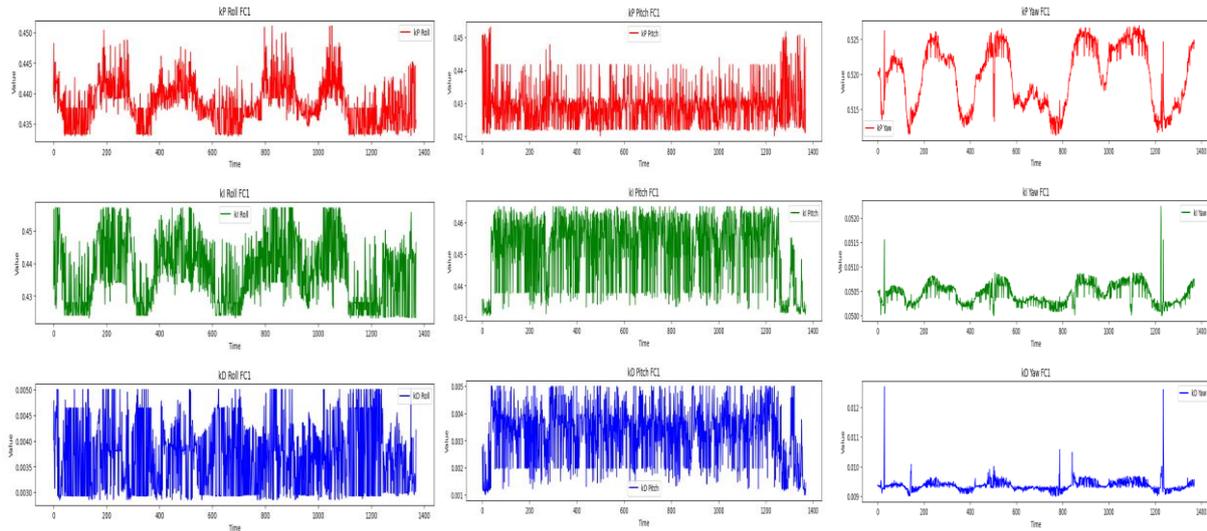


Рис. 63. Сравнение ожидаемого и реального рыскания FC1

При этом изменения во времени коэффициентов усиления соответствующего регулятора для разных углов были следующие как приведено на рис. 64.



(а) (б) (в)  
 Рис. 64. График коэффициентов усиления FC1 (крен (а), тангаж (б) и рыскание (в))

Далее представлены графики отклонений от целевого сигнала для второго нечёткого регулятора FC2. Рисунок 65 демонстрирует сравнение поведения ожидаемого и реального угла крена при FC2.

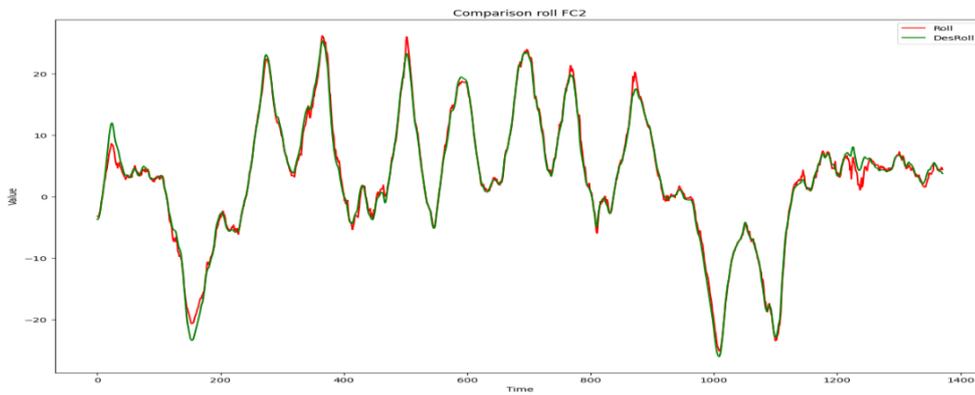


Рис. 65. Сравнение ожидаемого и реального угла крена при FC2

Рисунок 66 представляет сравнение ожидаемого и реального угла тангажа FC2.

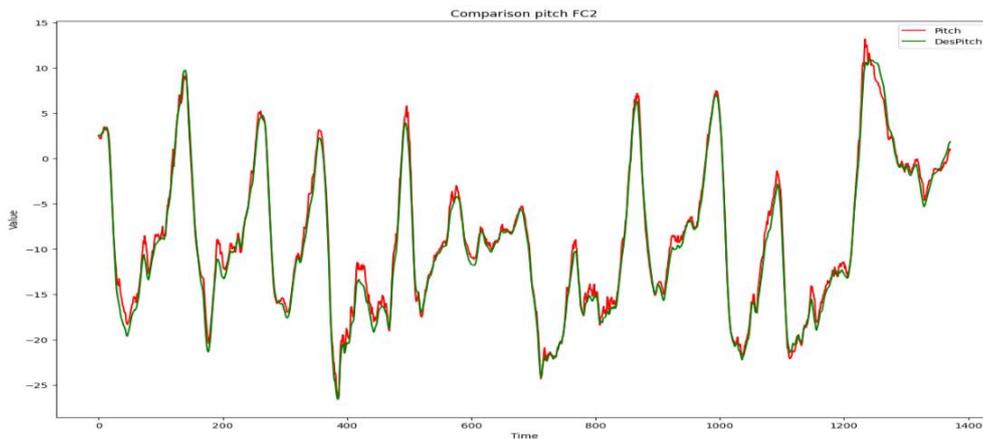


Рис. 66. Сравнение ожидаемого и реального угла тангажа при FC2

Рисунок 67 представляет сравнение ожидаемого и реального угла рыскания при FC2.

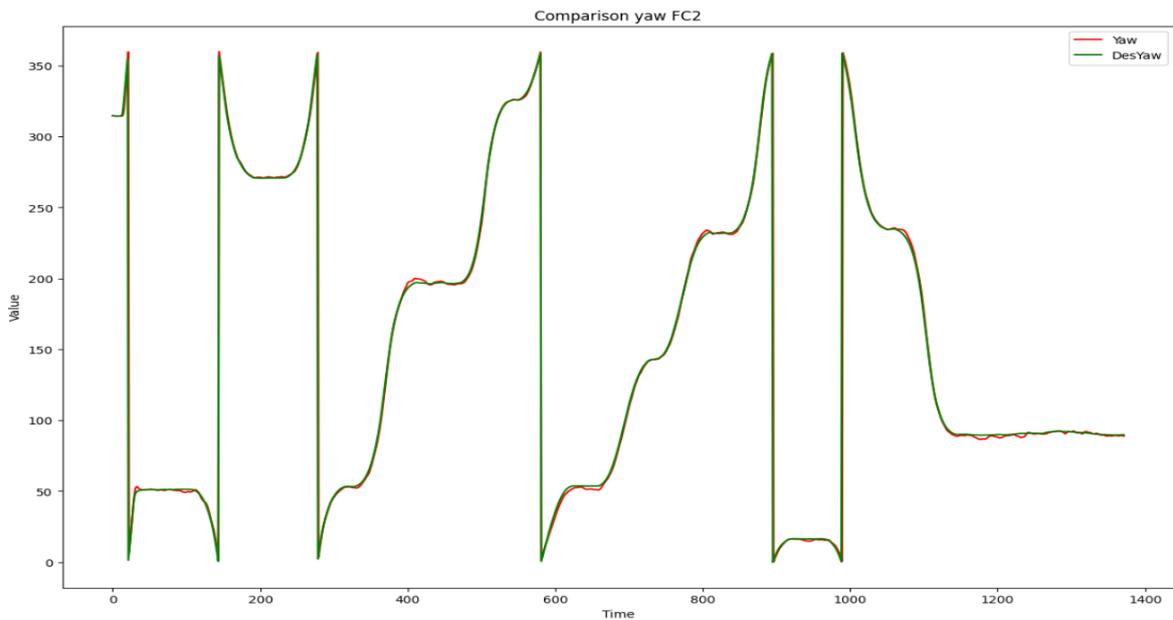


Рис. 67. Сравнение ожидаемого и реального углов рыскания при FC2

Рисунок 68 представляет результаты поведения коэффициентов усиления нечеткого регулятора FC2 для различных углов БС.

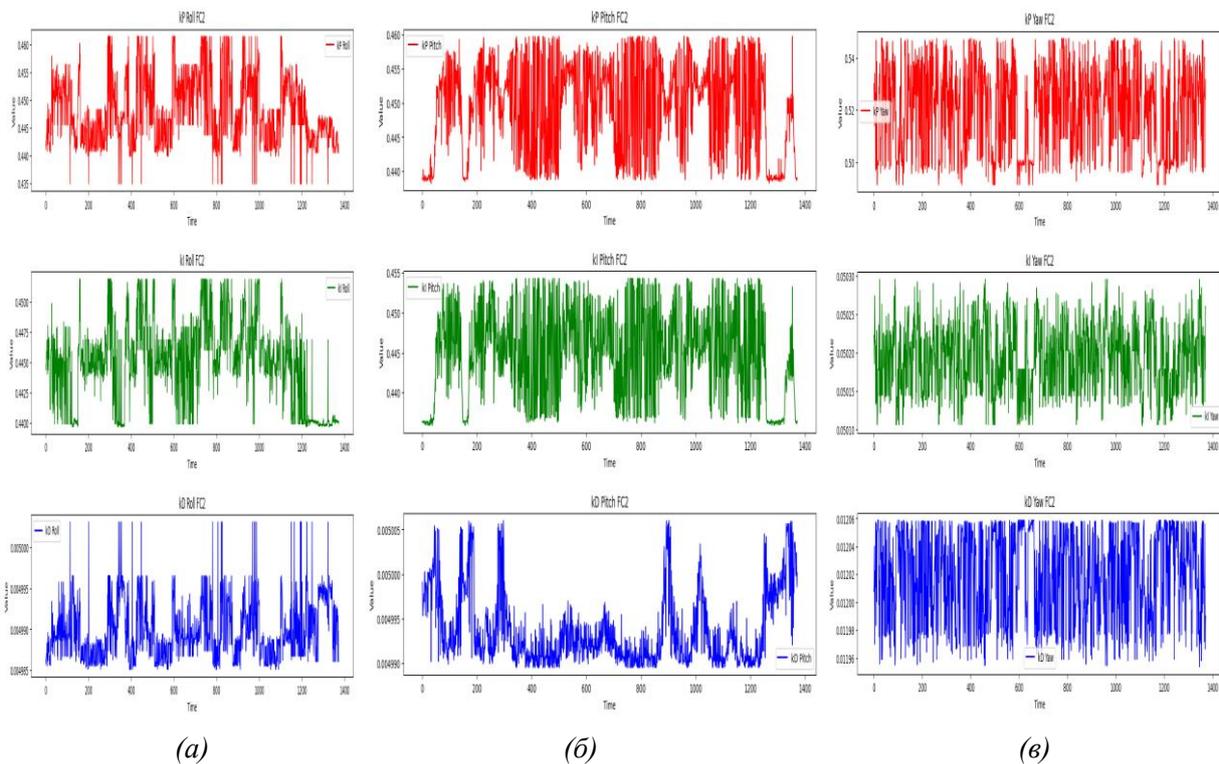


Рис. 68. График коэффициентов усиления нечеткого регулятора в ситуации С2 (крен (а), тангаж (б) и рыскание (в))

Теперь рассмотрим функционирования системы в ситуации С2 на основе квантового регулятора. Сравнение ожидаемого и реального углов крена при QFI приведено на рис. 69.

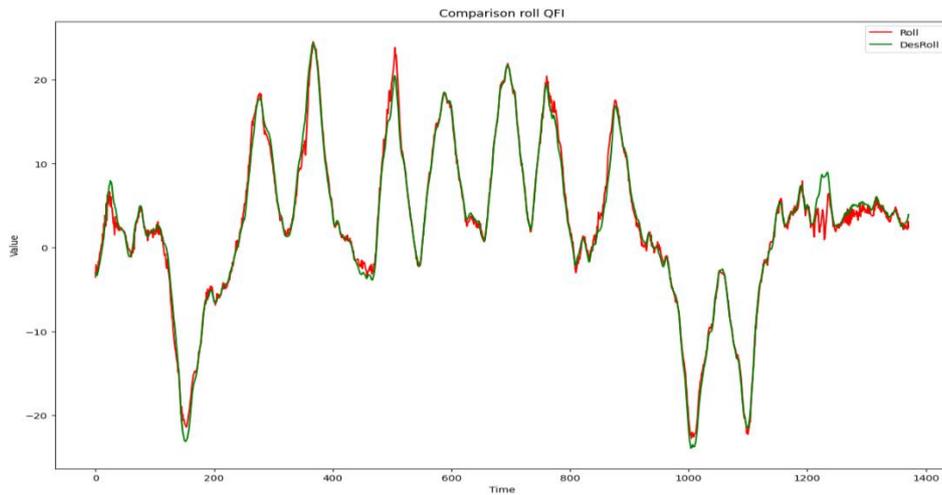


Рис. 69. Сравнение ожидаемого и реального углов крена при QFI

Сравнение ожидаемого и реального углов тангажа при QFI приведено на рис. 70.

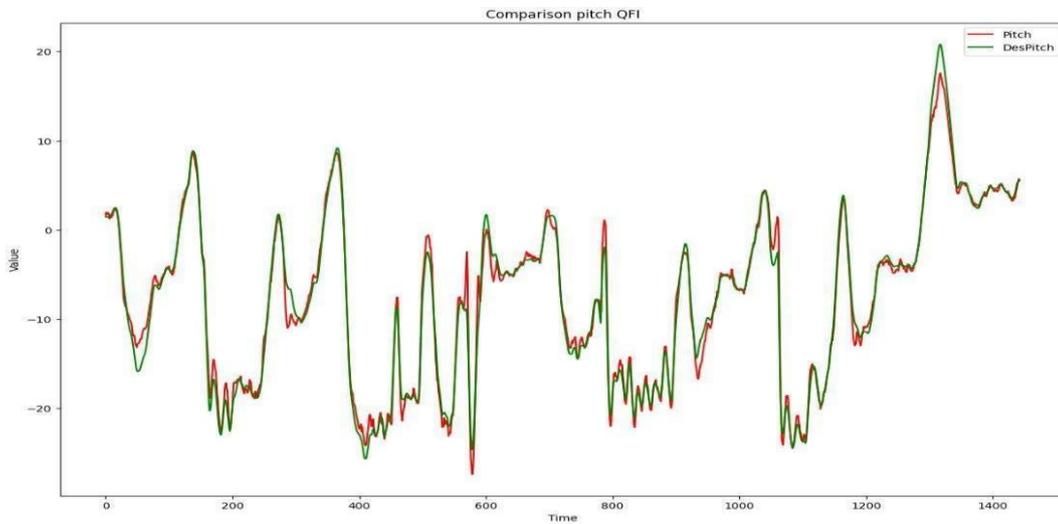


Рис. 70. Сравнение ожидаемого и реального углов тангажа при QFI

Сравнение ожидаемого и реального углов рыскания при QFI приведено на рис. 71.

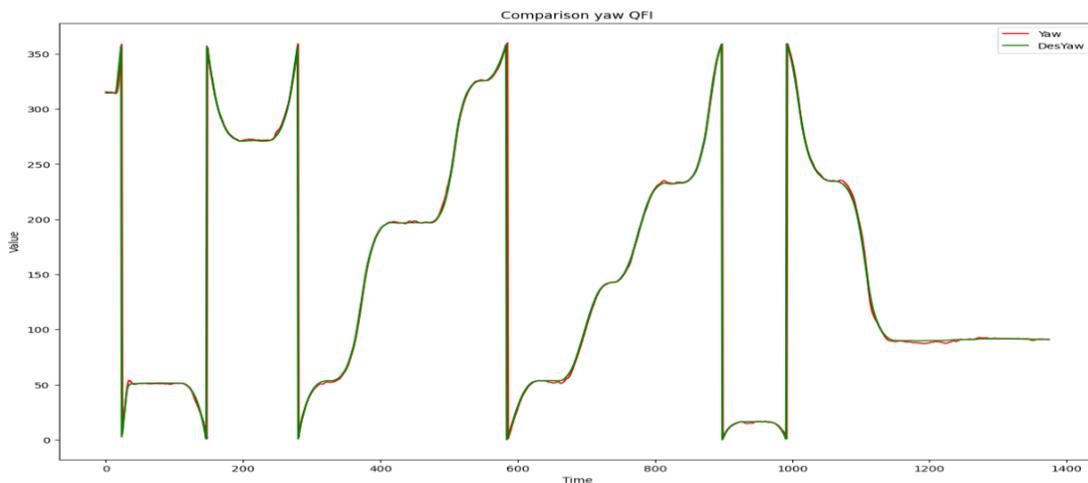


Рис. 71. Сравнение ожидаемого и реального углов рыскания при QFI.

Рисунок 72 демонстрирует поведение коэффициентов усиления квантового регулятора при  $QFI$  (угол крена).

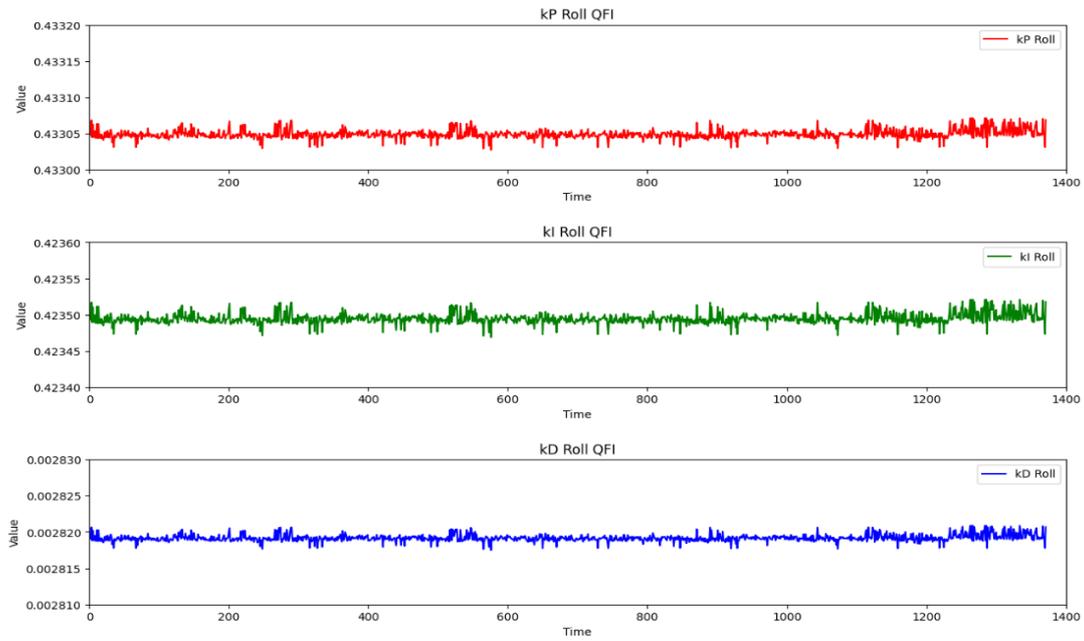


Рис. 72. График коэффициентов усиления при  $QFI$  (крен)

Рисунок 73 демонстрирует поведение коэффициентов усиления квантового регулятора при  $QFI$  (угол тангажа).

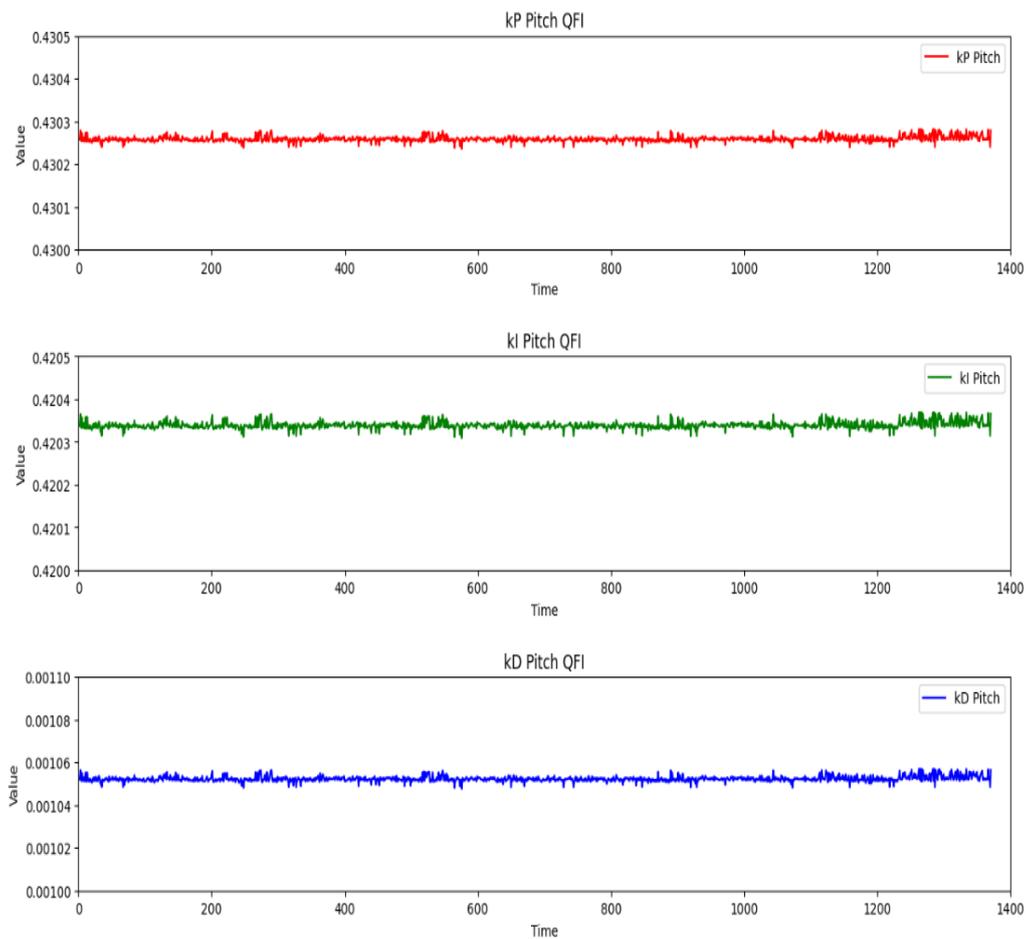


Рис. 73. График коэффициентов усиления  $QFI$  (Тангаж)

Рисунок 74 демонстрирует поведение коэффициентов усиления квантового регулятора при  $QFI$  (угол рыскания).

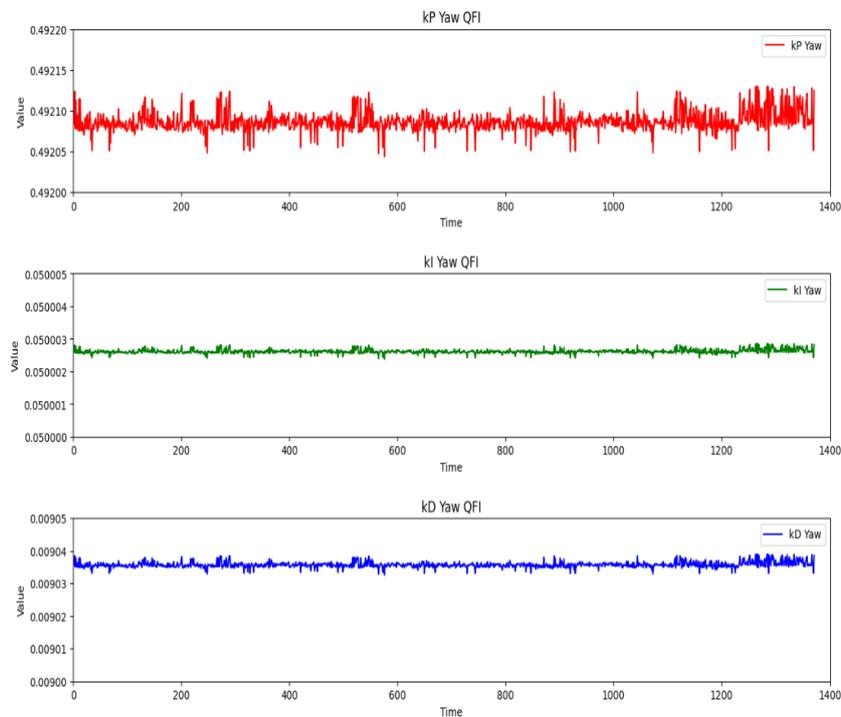


Рис. 74. График коэффициентов усиления при  $QFI$  (Рыскание)

Рассмотрим прямые и косвенные показатели качества управления для второй ситуации управления  $S2$ .

Рисунок 75 демонстрирует сравнение поведения ошибки отклонения по углу крена с  $QFI$  и других регуляторов.

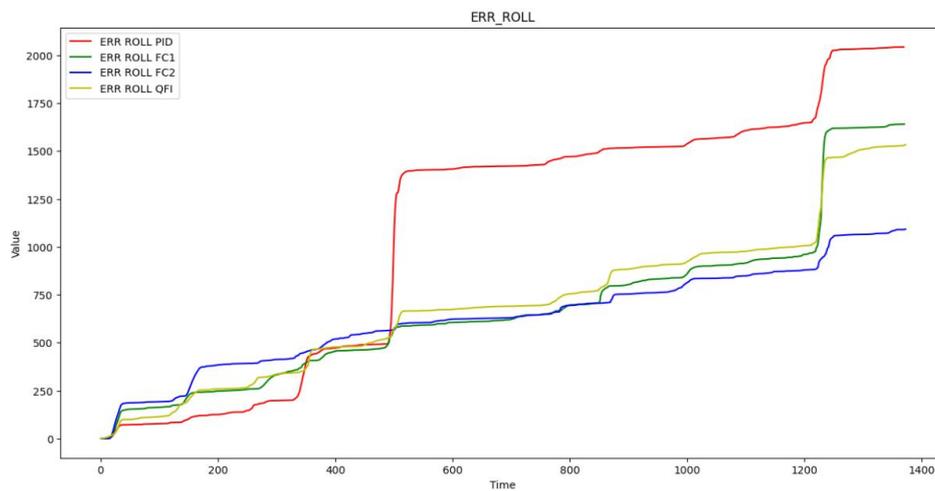


Рис. 75. График ошибки крена с  $QFI$

Из рис. 75 видно, что нечеткий регулятор  $FC2$  имеет наименьшую квадратичную интегральную ошибку по крену. Квантовый и нечеткий  $FC1$  регуляторы показывают меньшую ошибку, чем ПИД – регулятор.

Рисунок 76 демонстрирует сравнение поведения ошибки отклонения по углу тангажа с  $QFI$  и других регуляторов.

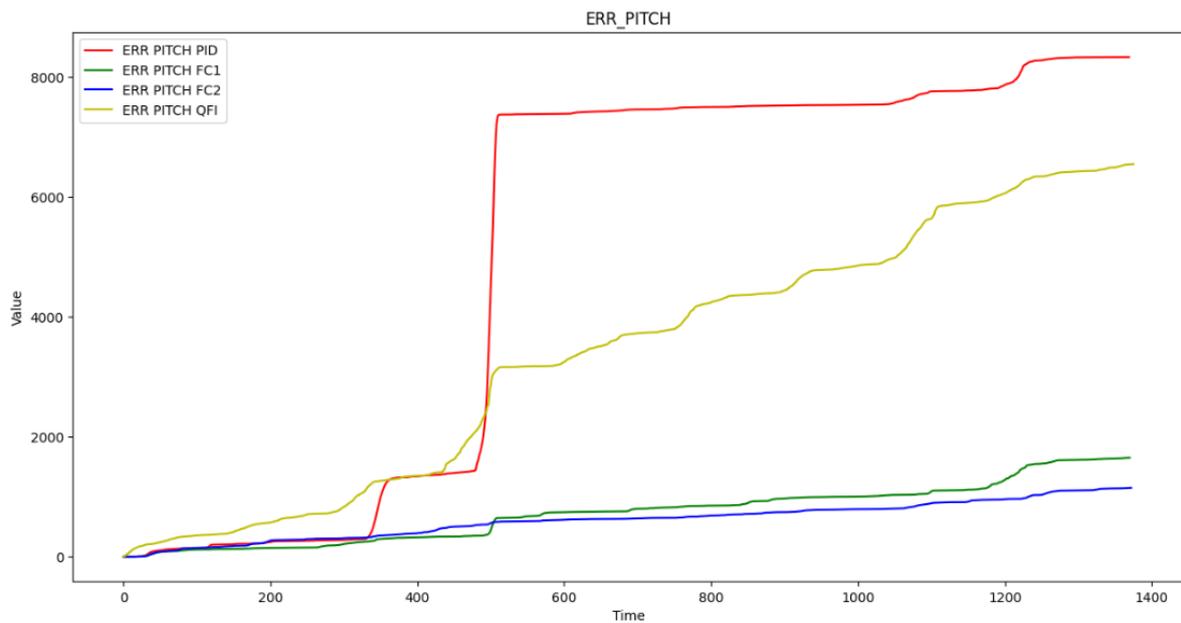


Рис. 76. График ошибки тангажа с *QFI*

Из рис. 76 видно, что нечеткие регуляторы имеют наименьшую квадратичную интегральную ошибку по тангажу. Квантовый регулятор показывает меньшую ошибку, чем ПИД – регулятор.

Рисунок 77 демонстрирует сравнение поведения ошибки отклонения по углу рыскания с *QFI* и других регуляторов.

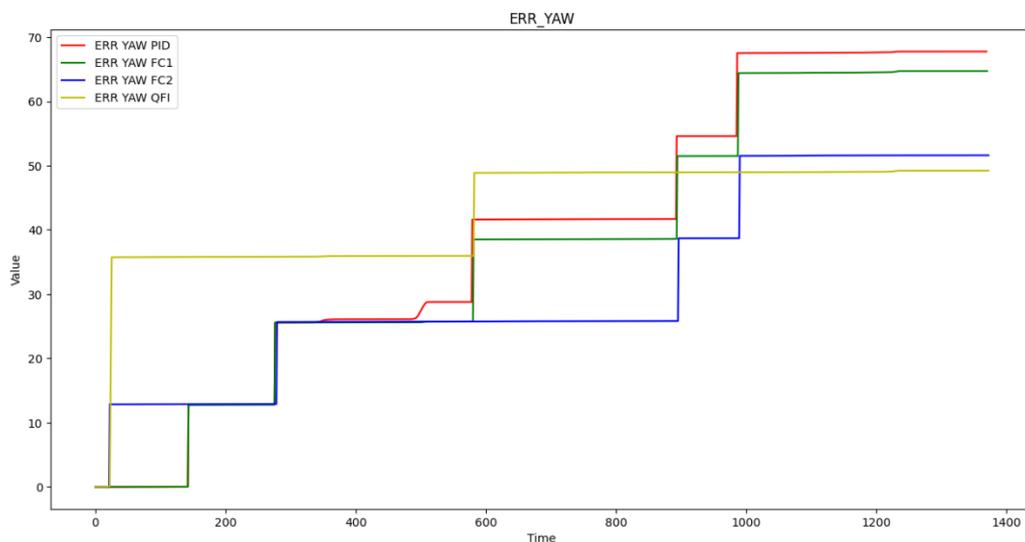


Рис. 77. График ошибки рыскания с *QFI*

Из рис. 77 видно, что квантовый регулятор имеет наименьшую квадратичную интегральную ошибку по углу рыскания. Нечеткие регуляторы показывают меньшую ошибку, чем ПИД – регулятор.

Таблица 5 представляет сравнительные характеристики показателей перерегулирования как одного из важных качеств управления.

Табл. 5. Показатель перерегулирования управления

Тип регулятора	Крен	Тангаж
ПИД	10.72	21.04
НР1	9.5	16.73
НР2	12.18	21.85
КНВ	<b>7.51</b>	<b>18.8</b>

Как следует из табл. 5 следует что квантовый регулятор обладает лучшими качествами управления по сравнению с другими регуляторами.

## 6. Сравнение энтропийных показателей качества управления

Производство и скорость производства энтропии регулятора характеризуют потерю полезного ресурса регулятором и влияют на перераспределение качеств управления таких как устойчивость, управляемость и робастность [15].

### 6.1. Фазовые энтропийные портреты и динамического поведения регуляторов.

На рис. 78 приведено описание фазовых портретов ошибки управления динамического поведения регуляторов.

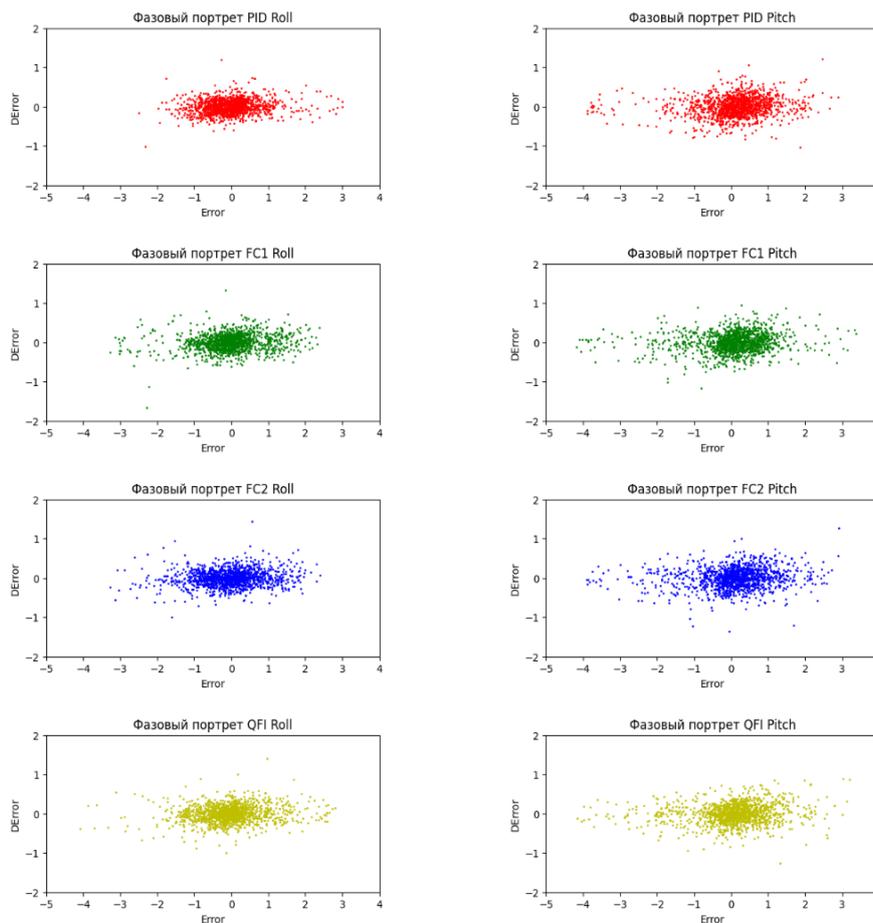


Рис. 78. Фазовые портреты системы в ситуации С1

На рис. 79 приведено описание фазовых портретов динамического поведения энтропии регуляторов в ситуации С1.

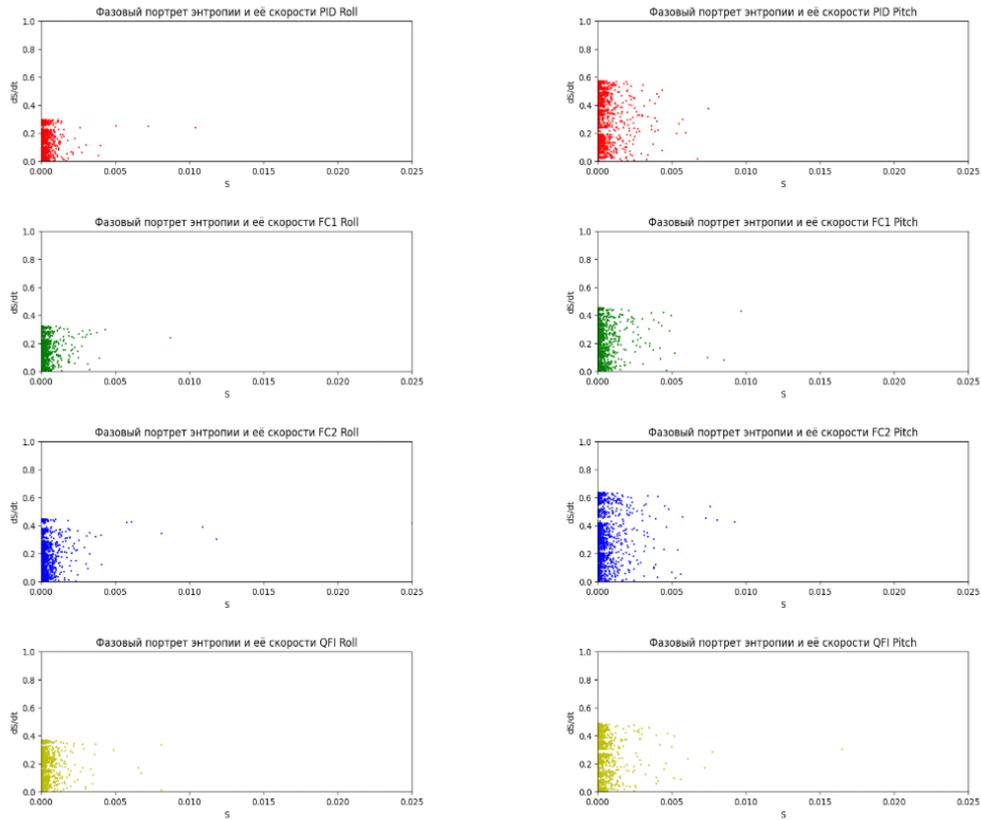


Рис. 79. Фазовые портреты энтропии в ситуации С1

Интегральные показатели роста энтропии

Интегральные показатели производства энтропии регуляторов приведены на рис. 80.

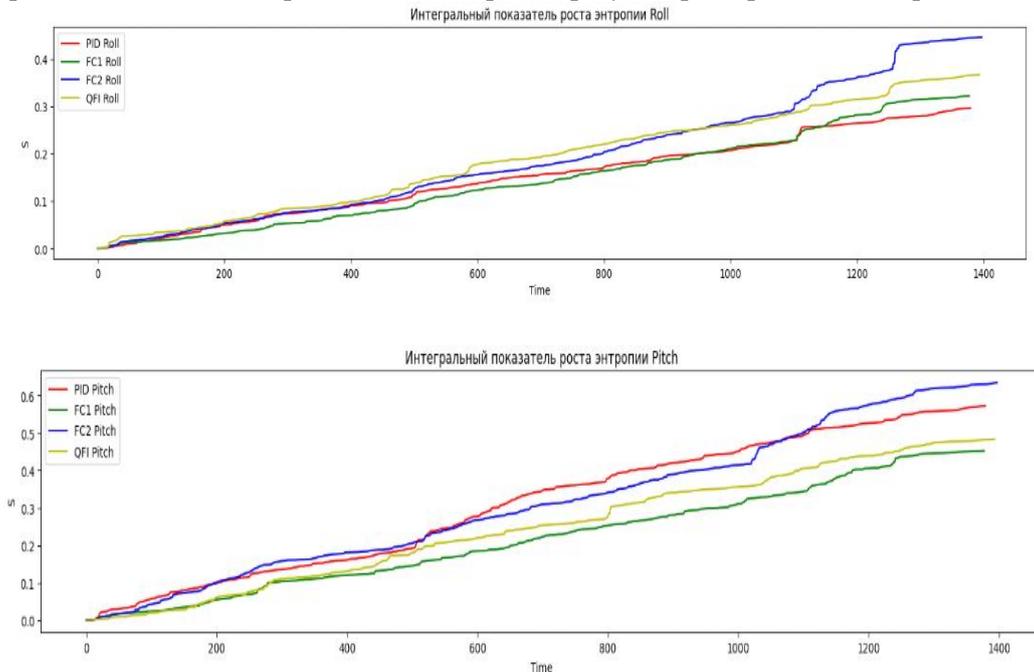


Рис. 80. Интегральные показатели производства энтропии

Изменение во времени коэффициентов усиления интеллектуальных регуляторов приведено для сравнения на примере управления углом крена, тангажа и рыскания на рис. 81 – 83.

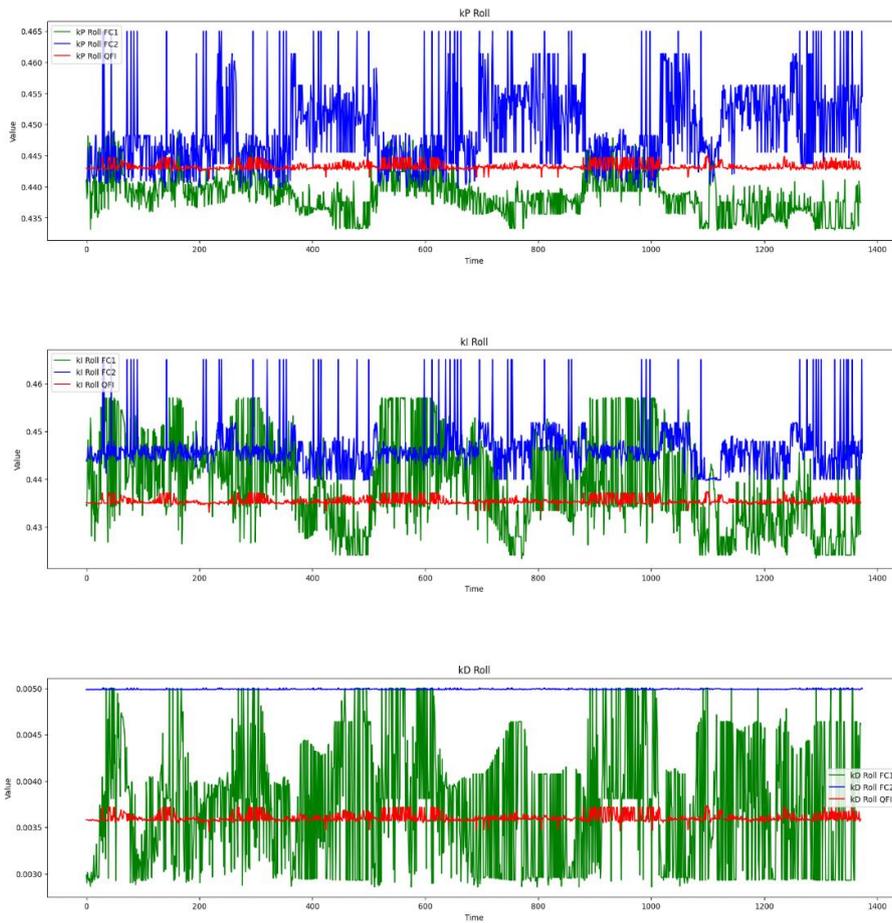


Рис. 81. Коэффициенты усиления интеллектуальных регуляторов для угла крена

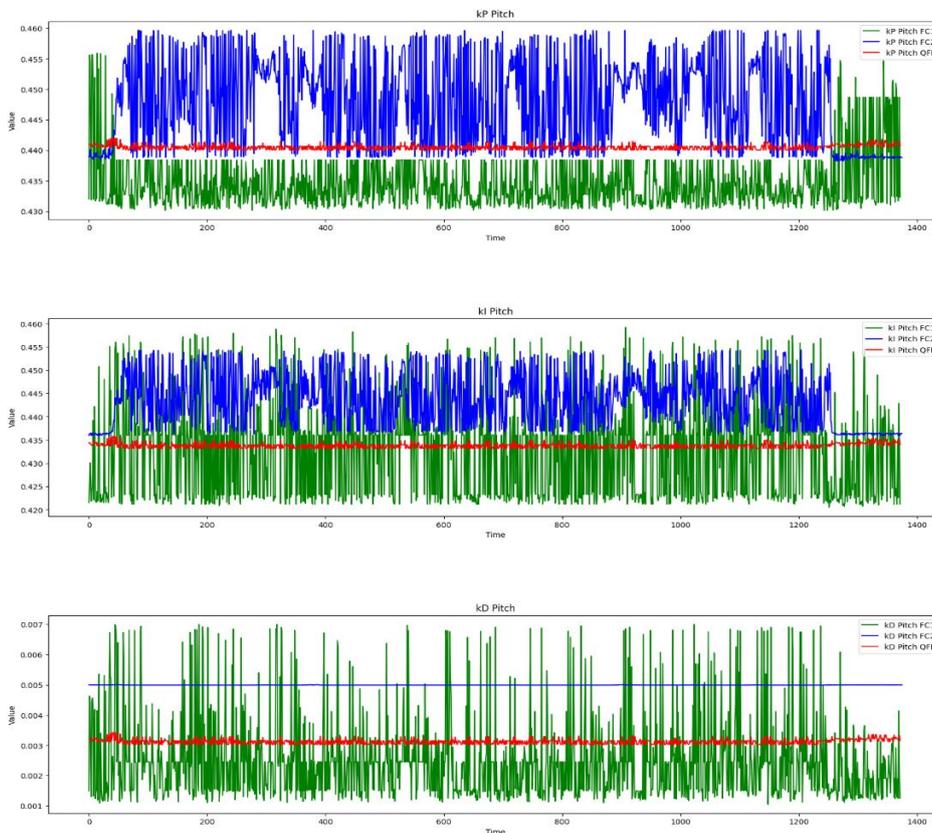


Рис. 82. Коэффициенты усиления интеллектуальных регуляторов для угла тангажа

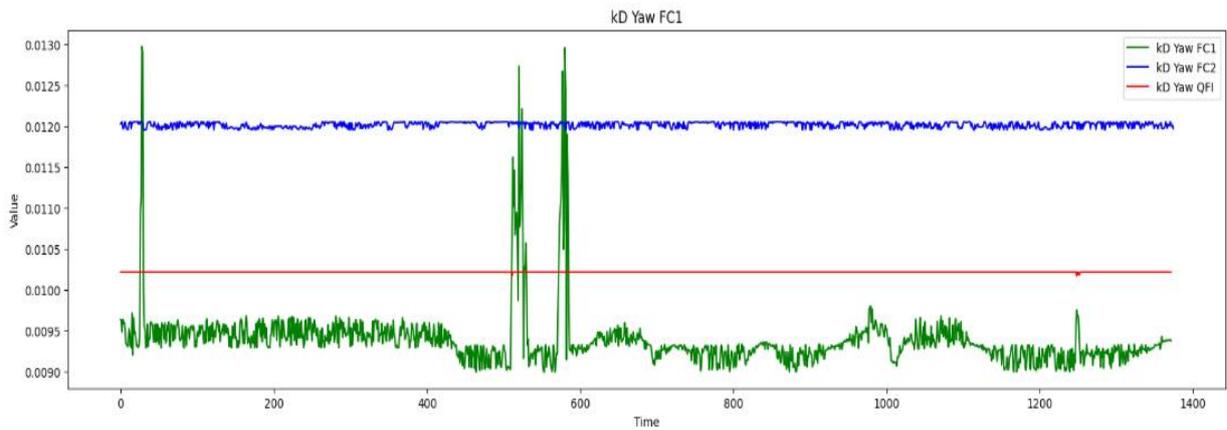
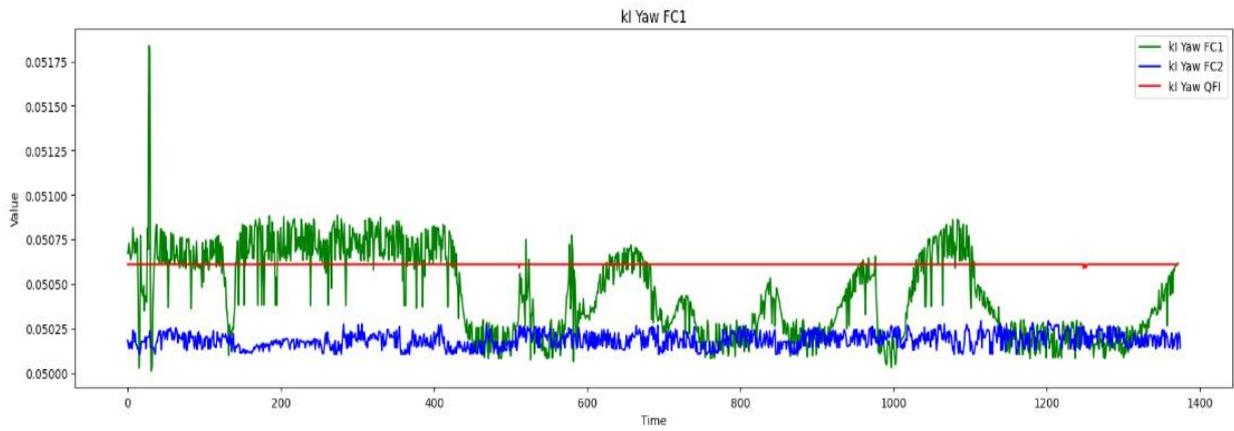
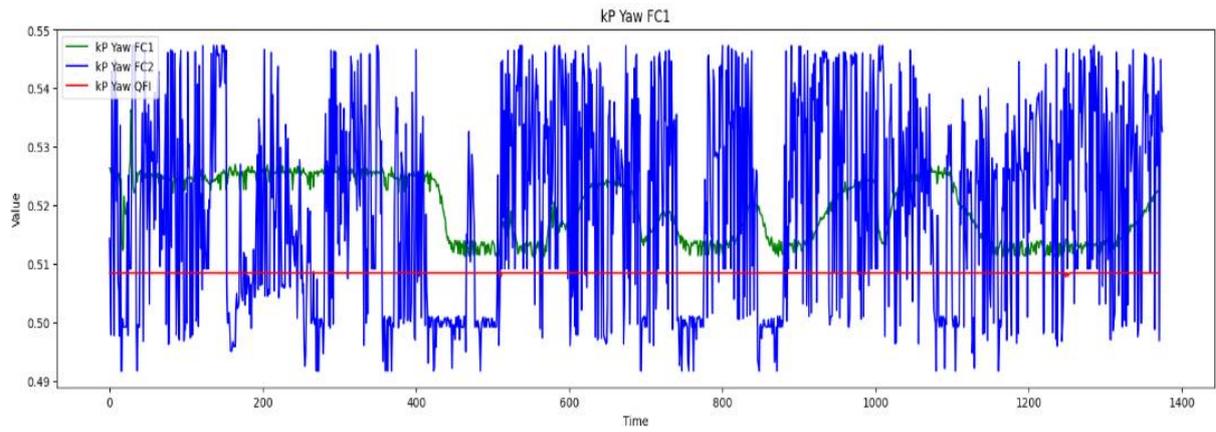


Рис. 83. Коэффициенты усиления интеллектуальных регуляторов угла рыскания

## 6.2. Сравнение энтропийных показателей в ситуации С2

На рис. 84 приведены фазовые портреты применяемых регуляторов в ситуации С2.

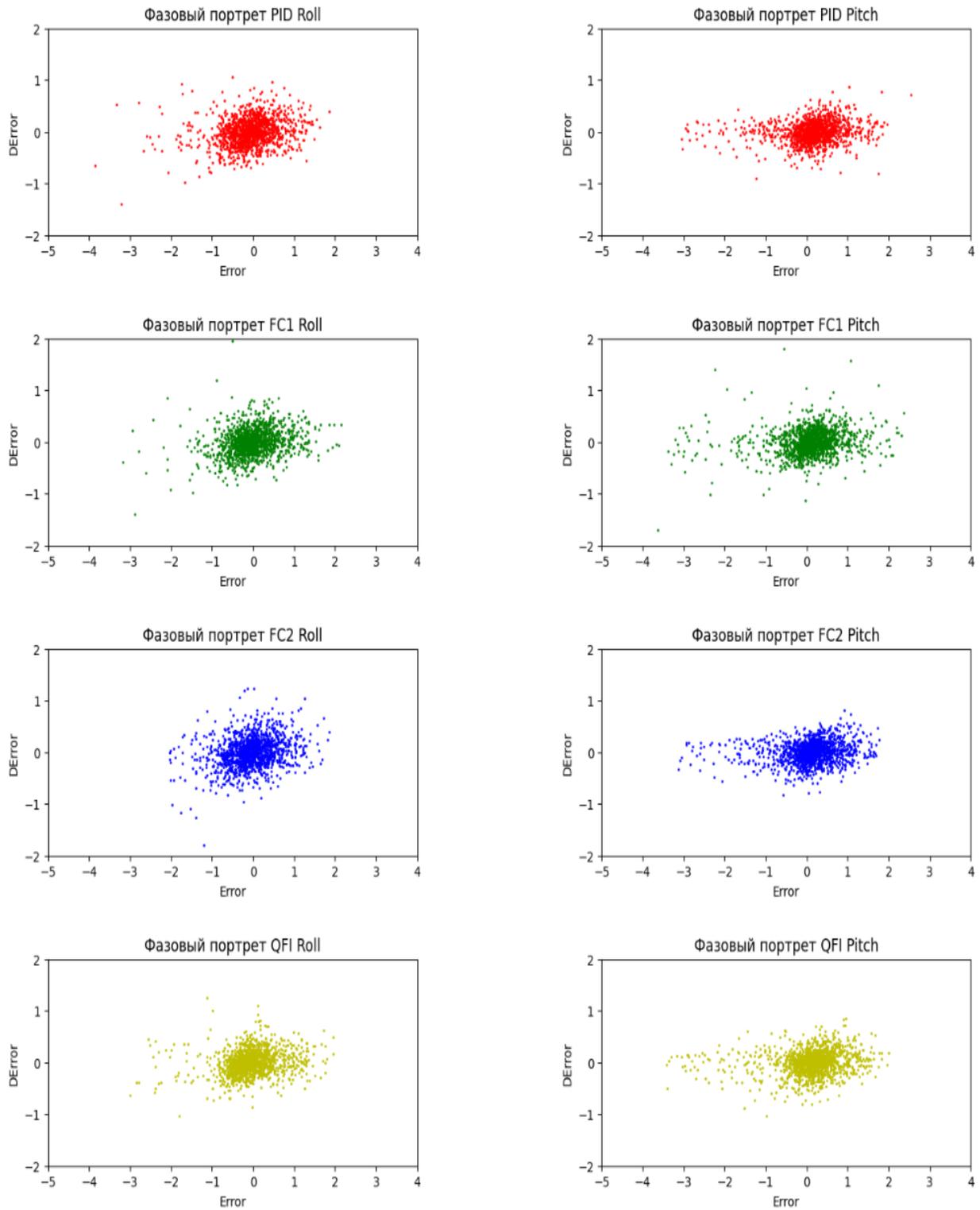


Рис. 84. Фазовые портреты регуляторов в ситуации C2

Энтропийные характеристики регуляторов приведены на рис. 85 и 86.

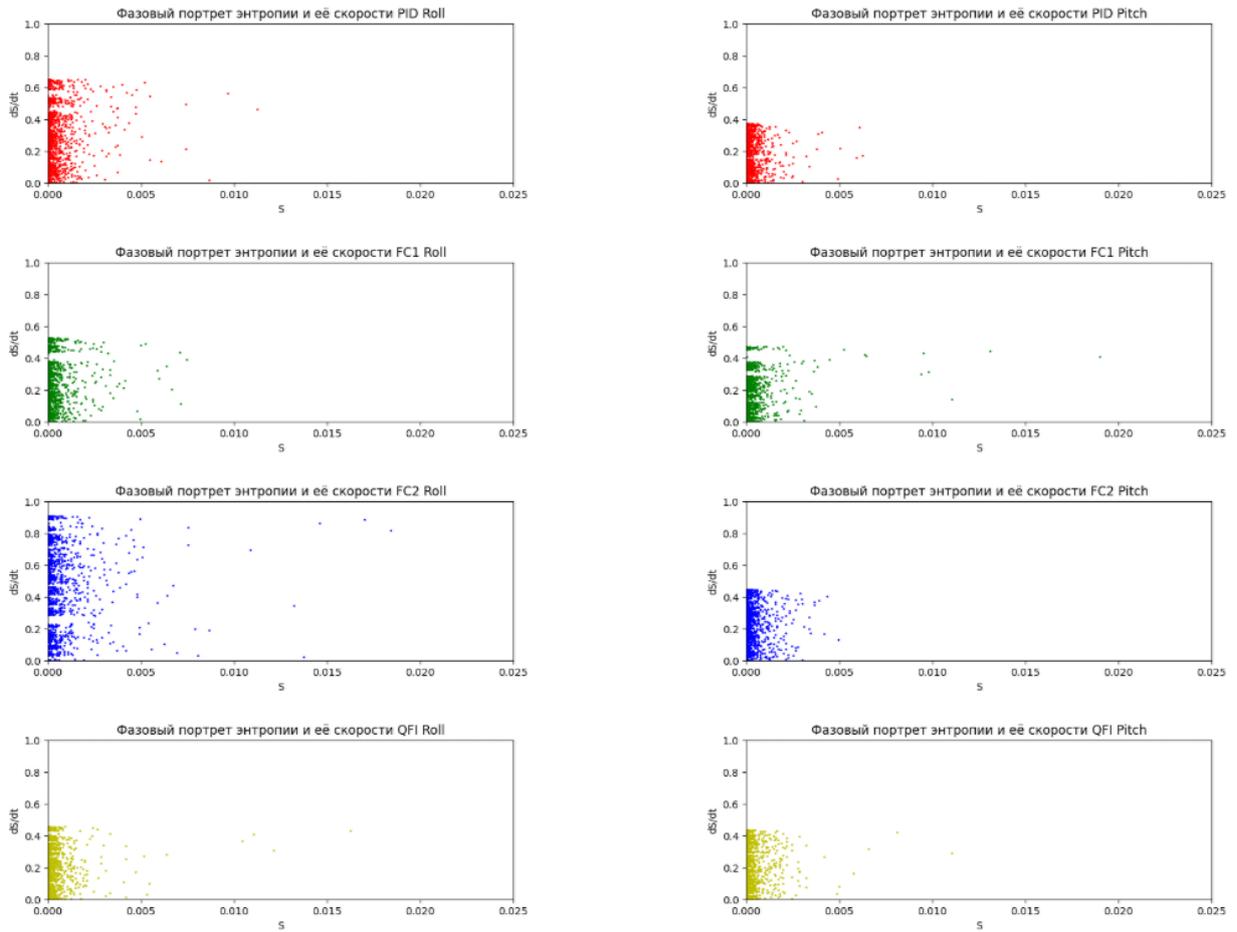


Рис. 85. Фазовые портреты энтропии в ситуации С2

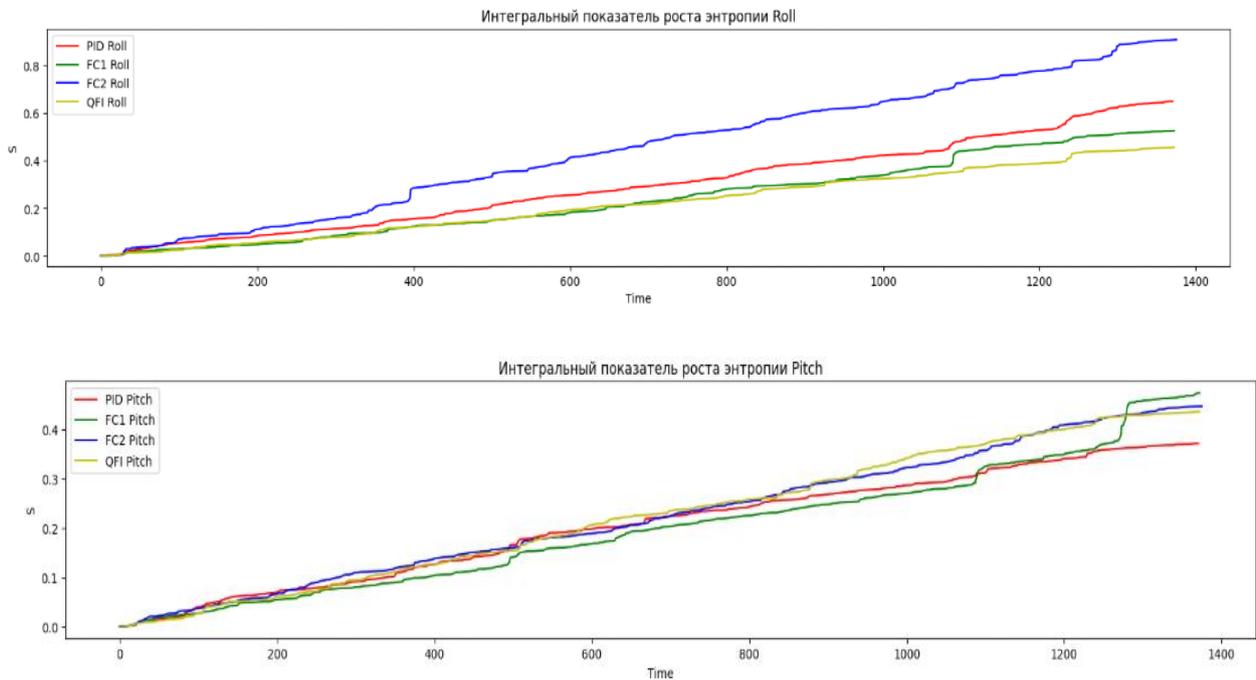


Рис. 86. Интегральные показатели производства энтропии регуляторами в ситуации С2 для углов крена и тангажа

Сравнение динамики изменения коэффициентов усиления регуляторов приведено на рис. 87 – 89.

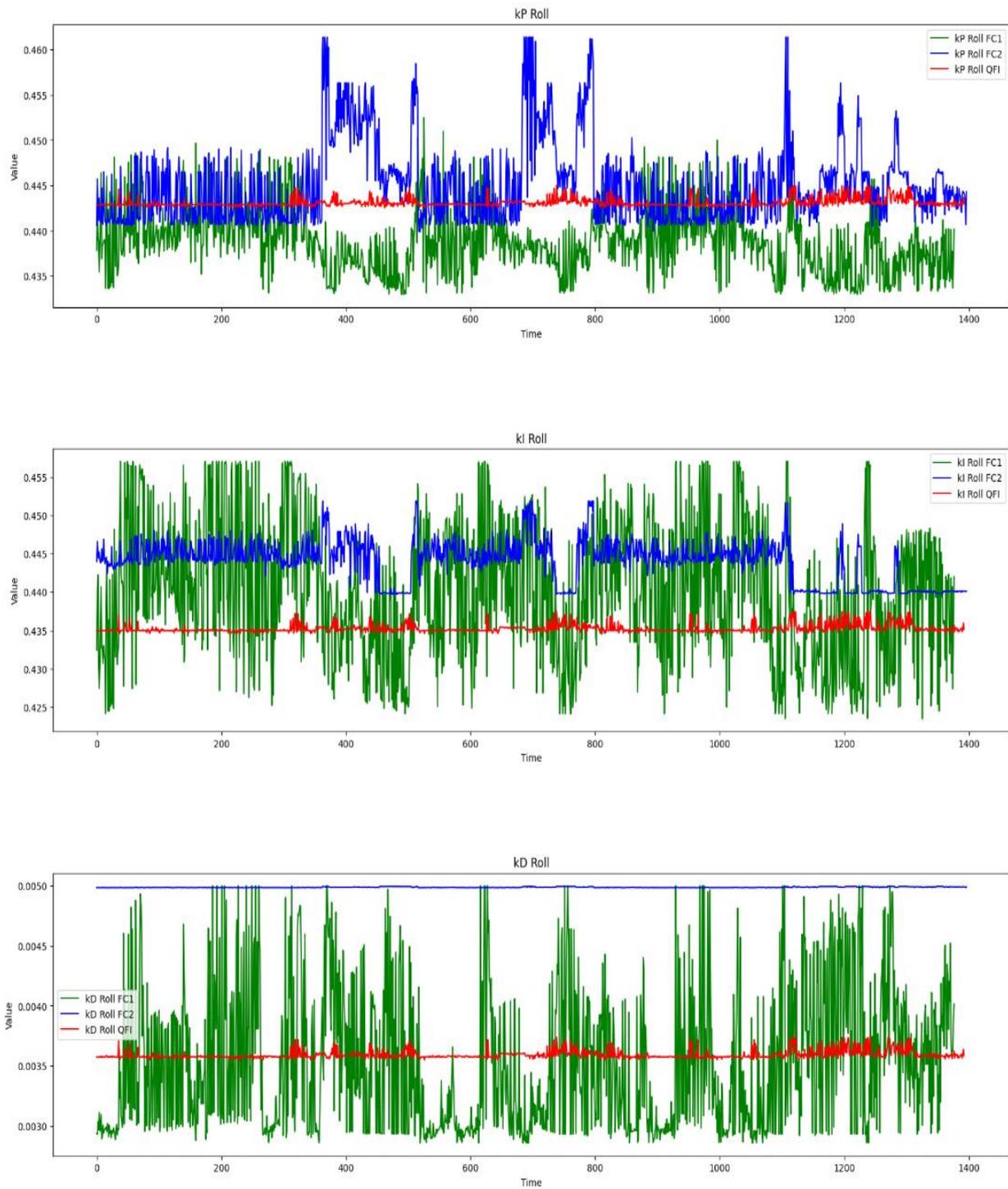


Рис. 87. Коэффициенты усиления интеллектуальных регуляторов

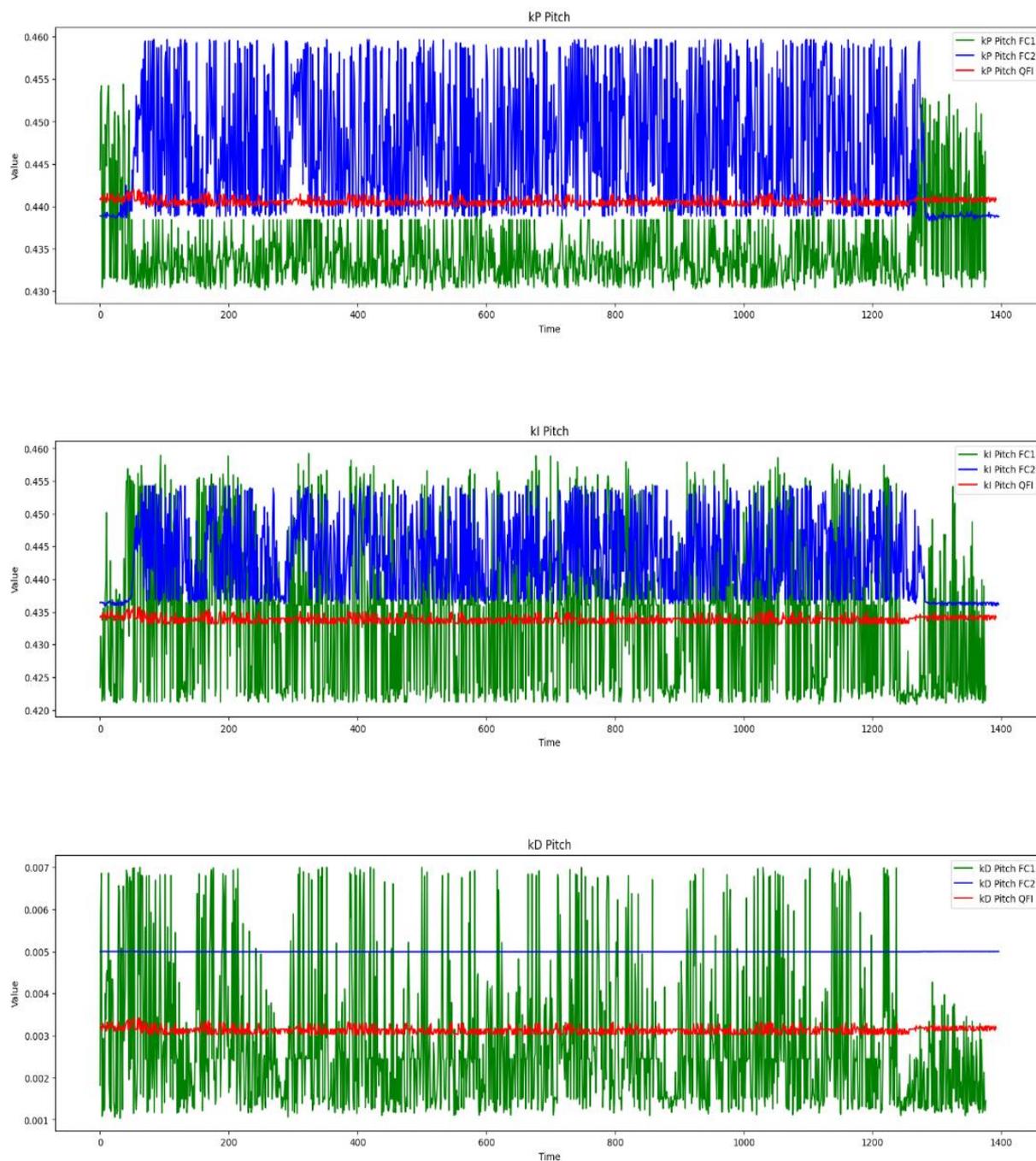


Рис. 88. Коэффициенты усиления интеллектуальных регуляторов

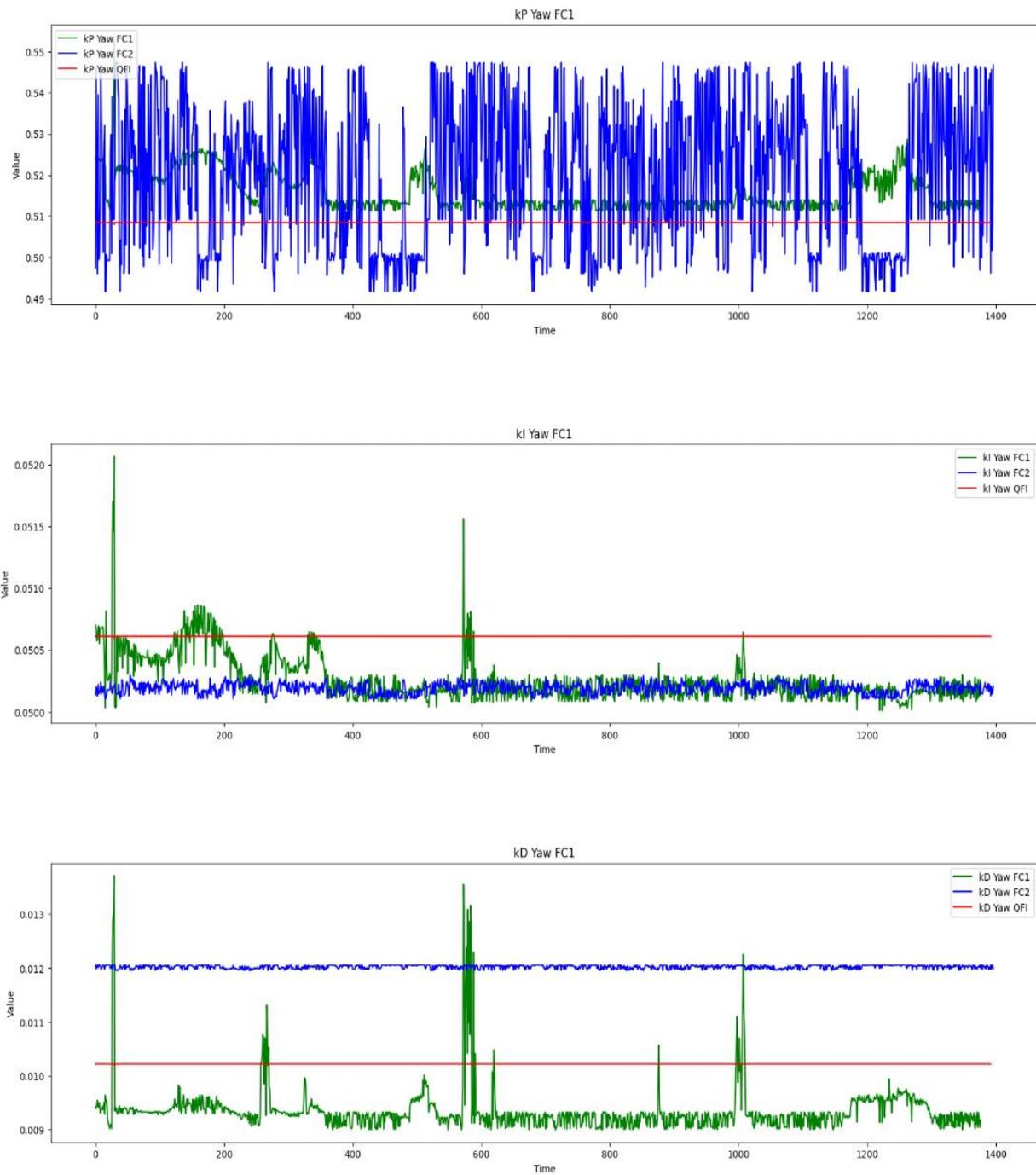


Рис. 89. Коэффициенты усиления интеллектуальных регуляторов

Из результатов, приведенных на рис. 87 – 89 отчетливо видно, что квантовый регулятор обладает простой физической реализацией изменением во времени коэффициентов усиления, квантовый нечеткий вывод редуцирует избыточность информации в законах изменения коэффициентов усиления нечетких регуляторов, квантовый регулятор обладает способностью рационального расхода полезного ресурса для гарантированного достижения цели управления.

При этом реализуется синергетический информационный эффект: из неточных баз знаний нечетких регуляторов формируется робастный квантовый регулятор.

## Результаты проведённого исследования

По итогам проделанной работы получены следующие результаты:

Разработана и апробирована технология проектирования встраиваемых интеллектуальных систем управления (ИСУ) для беспилотных воздушных судов (БВС).

Разработаны и реализованы программные компоненты, реализующие процесс проектирования ИСУ. В частности, генетические алгоритмы для извлечения обучающих сигналов с математической модели (*Gazebo*) и реального объекта интеллектуализации. Нечеткие нейросетевые контроллеры для управления БВС. Квантовые контроллеры для управления БВС. Инструментарии на мягких и квантовых вычислениях для создания встраиваемых структур интеллектуальных контроллеров.

Составлены и апробированы полетные программы обучающих и тестирующих испытаний. Подготовлены скрипты обучения, маршруты испытания и аппаратная поддержка полетов.

По результатам применения ИСУ для управления БВС достигнуты следующие показатели качества управления:

В типовой ситуации управления нечеткие регуляторы показывают явное преимущество. Поддержание высоты улучшилось на 10%. Ошибка по тангажу снизилась на 17%. Ошибка по крену снизилась на 8%.

В типовой ситуации управления квантовые регуляторы обладают низкой сложностью выполненных команд и высоким качеством управления. В частности, сложность управления снизилась на 8%. Относительно классического ПИД – регулятора поддержание высоты улучшилось на 3%. Ошибка по тангажу снизилась на 5%. Однако квантовый регуляторов уступает НР на 12% по тангажу. Ошибка по крену снизилась на 5%.

В типовой ситуации показатель перерегулирования нечеткого контроллера относительно ПИД – регулятора уменьшился на 49% по крену и на 42% по тангажу. Для квантовых регуляторов показатель перерегулирования относительно ПИД – регулятора уменьшился на 73% по крену и на 29% по тангажу.

В непредвиденной ситуации управления: поддержание высоты улучшилось также на 10%. Ошибка по тангажу снизилась на 86%. Ошибка по крену снизилась на 46%. Ошибка по рысканию снизилась на 24%.

В непредвиденной ситуации управления квантовые регуляторы сложность управления снизили на 10%. Относительно классического ПИД– регулятора ошибка по тангажу снизилась на 21%. Однако квантовый регуляторов уступает НР, разработанного для управления с полезной нагрузкой на 65%. Ошибка по крену относительно ПИД снизилась на 25%, ошибка по рысканию снизилась на 27%.

В непредвиденной ситуации показатель перерегулирования нечеткого контроллера относительно ПИД – регулятора уменьшился на 11% по крену и на 20% по тангажу. Для квантовых регуляторов показатель перерегулирования относительно ПИД – регулятора уменьшился на 30% по крену и на 11% по тангажу.

Квантовый регулятор во всех ситуациях управления расходует меньше полезного ресурса и имеет больший запас устойчивости.

В перспективе, возможно расширить применение математического моделирование в процессе обучения ИСУ для БВС. Это позволит включать в систему управления более сложные траектории и разнообразные ситуации обучения в том числе и нештатные. Потенциал квантового управления раскрыт не окончательно. Необходимо исследовать возможность применения различных корреляций между регуляторами. Так же возможно расширение технологии и на другие регуляторы системы управления. Регуляторы поддержания высоты, регуляторы торможения и т.д. Более глубокая проработка и изучение базовой прошивки позволит выявить использование ограничительных параметров (допустимых углов и скоростей) в системе управления. Возможна проработка выполнения более сложных траекторий полета, в частности выполнения таких фигур пилотажа как («бочка», «петля» и т.д.)

## Список источников

1. Воздушный кодекс Российской Федерации : Федеральный закон №60-ФЗ : принят Государственной Думой 19.02.1997 : текст с изменениями и дополнениями на 08.08.2024.
2. Число учтенных в России дронов выросло на треть // Федеральное агентство воздушного транспорта. — Официальный Интернет-ресурс Федерального агентства воздушного транспорта, 2010–2025. — Дата публикации: 18 января 2024. — URL: <https://favt.gov.ru/novosti-povosti/?id=10676>.
3. Аналитика отрасли беспилотной авиации // Аэронет 2035 : [Сайт Национальной технологической инициативы]. — АНО «Платформа НТИ», 2025. — URL: <https://xn--2035-43d4a7chr0j.xn--p1ai/dashboard>.
4. Терёхина, Е. Е. К вопросу о системах управления беспилотными воздушными судами // Молодой ученый. — 2023. — № 12 (459). — С. 16-19.
5. Плеханов П. А., Роенков Д. Н. Характеристики, возможности и проблемы использования беспилотных летательных аппаратов на железнодорожном транспорте // 79-я научно-техническая конференция Санкт-Петербургского НТО РЭС им. А.С. Попова, посвященная Дню радио. — 2024. — С. 247-249.
6. Малясов А. А. Пути разработки системы управления для беспилотного транспортного вертолета // Молодой ученый. — 2020. — № 50 (340). — С. 66-69.
7. Перспективы совершенствования автоматизированных систем, контролирующих полеты, навигацию, посадку и связь в государственной авиации / А. С. Варламов, Д. Р. Байрамгалин, Д. А. Кокшаров // Молодой ученый. — 2023. — № 48 (495). — С. 32-35.
8. Зинченко Д. С., Нерубенко А. С. Беспилотные воздушные судна в деятельности органов внутренних дел: современное состояние и перспективы развития // Вестник Белгородского юридического института МВД России имени И.Д. Путилина. — 2024. — №2. — С. 42-49.
9. Lee R. Betaflight, ArduPilot and INAV, which FC Firmware is right for you? — Get FPV drone kit Tutorials and Reviews. 2025. — URL: <https://www.mepsking.com/blog/betaflight-ardupilot-inav-fc-firmware.html>.
10. ArduPilot Copter — Copter Documentation // ArduPilot Documentation. — ArduPilot Dev Team, 2024. — URL: <https://ardupilot.org/copter/>.
11. Rotella N. ArduCopter Flight Controllers. — URL: <https://nrotella.github.io/journal/arducopter-flight-controllers.html>.
12. Litvintseva L. V., Ulyanov S. V., Ulyanov S. S. Design of robust knowledge bases of fuzzy controllers for intelligent control of substantially nonlinear dynamic systems: II. A soft computing optimizer and robustness of intelligent control systems // Journal of computer and systems sciences international. — 2006. — Т. 45. — № 5. — С. 744 -771.
13. Интеллектуальная когнитивная робототехника. Часть 4.1. Квантовый «сильный» вычислительный интеллект в интеллектуальном управлении роботизированными автономными системами в «Индустрии 4.0/5.0» / Капков Р.Ю., Решетников А.Г., Тятюшкина О.Ю., Ульянов С.В. — Москва: Курс, 2025. — 483 с.
14. Intelligent cognitive robotics. Vol. 3: Quantum computational toolkit of quantum self-organized intelligent control system simulator: quantum deep machine learning on quantum-inspired neural network and quantum genetic algorithms / A. G. Reshetnikov, S. V. Ulyanov, P. V. Zrellov, D. P. Zrellova. — Kurs, 2023. — 432 с.
15. Ulyanov S. V. Intelligent cognitive robotics. Vol. 2. Quantum self-organization of imperfect knowledge bases: quantum intelligent force control and information-thermodynamic law of extracted informed useful work. — Kurs, 2022. — 472 с.
16. Боровинский В.В., Капков Р.Ю., Решетников А.Г., Ульянов С.В. Встраиваемый программный модуль, реализующий нечеткий контроллер для управления беспилотным летательным

аппаратом мультироторного типа. Свидетельство Государственной регистрации программы для ЭВМ. № 2024666917.

17. Свидетельство о государственной регистрации программы для ЭВМ № 2024666917 Российская Федерация. Встраиваемый программный модуль реализующий нечёткий контроллер для управления беспилотным летательным аппаратом мультироторного типа : № 2024666154 : заявл. 12.07.2024 : опубл. 17.07.2024 / В. В. Боровинский, М. С. Катулин, Р. Ю. Капков [и др.] ; заявитель Общество с ограниченной ответственностью «АТОМ». – EDN: TTGVID.
18. Свидетельство о государственной регистрации программы для ЭВМ № 2024680666 Российская Федерация. Встраиваемый программный модуль, реализующий квантовый нечёткий регулятор для управления беспилотным летательным аппаратом мультироторного типа : № 2024669806 : заявл. 22.08.2024 : опубл. 02.09.2024 / В. В. Боровинский, Р. Ю. Капков, А. Г. Решетников, С. В. Ульянов ; заявитель Общество с ограниченной ответственностью «АТОМ». – EDN: PTHLEK.