

Санкт-Петербургский государственный университет

*Мальцев Владимир Сергеевич*

Выпускная квалификационная работа

Разработка системы управления полетом  
БПЛА с использованием рандомизации в  
условиях навигации с существенными  
неопределенностями

Уровень образования: бакалавриат

Направление *09.03.04 «Программная инженерия»*

Основная образовательная программа *СВ.5080.2017 «Программная инженерия»*

Научный руководитель:  
к.ф.-м.н., Директор НОЦ СПбГУ "Математическая робототехника и искусственный  
интеллект" К.С. Амелин

Рецензент:  
к.т.н., ассистент университет ИТМО С.А. Вражевский

Санкт-Петербург  
2021

Saint Petersburg State University

*Vladimir Maltsev*

Bachelor's Thesis

Development of a UAV flight control system  
using randomization in navigation conditions  
with significant uncertainties

Education level: bachelor

Speciality *09.03.04 «Software Engineering»*

Programme *CB.5080.2017 «Software Engineering»*

Scientific supervisor:  
C.Sc., Konstantin Amelin

Reviewer:  
docent at ITMO, C.Sc., Sergey Vrazhevskii

Saint Petersburg  
2021

# Оглавление

<b>Введение</b>	<b>4</b>
<b>1. Постановка задачи</b>	<b>6</b>
<b>2. Обзор</b>	<b>7</b>
2.1. Алгоритмы отслеживания траектории БПЛА . . . . .	7
2.2. Методы фильтрации ошибок в управляющем сигнале . .	8
2.3. Методы измерения ветра на БПЛА с фиксированным кры- лом . . . . .	9
2.4. Адаптивный рандомизированный алгоритм . . . . .	10
2.5. Автопилот PX4 . . . . .	11
<b>3. Алгоритм оценки ветра</b>	<b>13</b>
3.1. Система координат . . . . .	13
3.2. Вычисление вектора ветра . . . . .	14
3.3. Алгоритм трубки Пито (РТА) . . . . .	15
3.4. Добавление рандомизации в управление БПЛА при неиз- вестных изменениях ветра . . . . .	18
3.5. Построение алгоритма оценки ветра . . . . .	20
<b>4. Особенности реализации модуля оценки ветра</b>	<b>22</b>
4.1. Архитектура автопилота . . . . .	22
4.2. Особенности реализации модуля оценки ветра . . . . .	24
<b>5. Тестирование и апробация</b>	<b>26</b>
5.1. Имитационное тестирование . . . . .	26
5.2. Апробация на реальном БПЛА . . . . .	29
<b>Заключение</b>	<b>31</b>
<b>Список литературы</b>	<b>33</b>

# Введение

В настоящее время беспилотные летательные аппараты (БПЛА) с фиксированным крылом являются предметом активных исследований. За счет того, что они используют подъемную силу, тем самым снижая расход батареи, такие БПЛА способны осуществлять автономные полеты на дальние расстояния и выполнять большой спектр задач. Для осуществления автономного полета для БПЛА задается маршрут в виде набора точек в пространстве, по которым выполняется навигация. Успешное выполнение прикладных задач, таких как проведение аэрофотосъемки местности или исследование магнитного поля участка земли, зависит от точности позиционирования и соблюдения заданных траекторий движения по маршруту.

Для решения задачи позиционирования используются показания с датчиков GPS в комбинации с инерциальной навигационной системой (ИНС). ИНС использует свойства инерции аппарата такие как линейное ускорение и угловую скорость для отслеживания и определения его положения в пространстве. Однако малые размеры БПЛА, с одной стороны, являются причиной их низкой цены по сравнению с пилотируемой авиацией, а с другой стороны, ограничением, из-за которого в БПЛА устанавливают небольшие ИНС. Такие ИНС со временем накапливают значительную погрешность, поэтому их применение ограничивается поддержанием текущего равновесия. Для позиционирования используются только данные с GPS, которые поступают в дискретные моменты времени и часто содержат погрешности. Также существенное влияние на полет оказывают динамические характеристики БПЛА и качество работы датчиков, установленных на нем. Вместе с тем существуют внешние случайные воздействия на БПЛА со стороны ветра, поэтому направление его движения и скорость могут колебаться.

Для сглаживания этих ошибок применяют различные линейные фильтры, среди которых наиболее типичными являются фильтр Калмана и его вариации [36]. Однако такие фильтры недостаточно хорошо справляются с ошибками, у которых математическое ожидание не равно ну-

лю [15]. К возникновению смещенной ошибки приводит также воздействие ветра. Одним из способов улучшить работу фильтра Калмана в такой ситуации является применение рандомизированных методов [14], в частности метода одновременно возмущаемой стохастической аппроксимации (Simultaneous Perturbation Stochastic Approximation, SPSA) [40]. При его использовании предполагается намеренно выводить систему из состояния равновесия с целью противодействия случайным воздействиям. В данном методе заложена математическая структура, позволяющая оценивать его стохастические свойства также для задач оптимизации, подверженных шуму и неопределенностям. Благодаря этим свойствам SPSA применяют для оптимизации многих задач адаптивного управления, например, управление группой БПЛА [5] и настройки сложных, крупномасштабных моделей [3].

В рамках данной дипломной работы предлагается расширить полученные в [4] результаты применения метода SPSA для противодействия случайным изменениям силы ветра, чтобы также минимизировать воздействие изменения направления ветра, а также разработать модуль оценки ветра, базирующийся на данном методе, для апробации на реальном БПЛА.

# 1. Постановка задачи

Целью работы является применение алгоритма SPSA для стабилизации полета БПЛА при навигации в условиях существенных неопределенностей — при случайных изменениях направления и скорости ветра. Для выполнения этой цели были поставлены следующие задачи.

1. Провести обзор предметной области.
2. Создать алгоритм оценки ветра, который включает рандомизированные поправки в управляющий сигнал.
3. Спроектировать и реализовать модуль включения рандомизированных поправок в управляющий сигнал автопилота PX4-Autopilot.
4. Провести имитационное моделирование и апробацию на реальном БПЛА самолетного типа.

## 2. Обзор

В обзоре будут рассмотрены алгоритмы отслеживания траектории и фильтрации ошибок, методы расчета ветра при полете на БПЛА с фиксированным крылом, а также метод SPSA и автопилот, на базе которого планируется тестирование.

### 2.1. Алгоритмы отслеживания траектории БПЛА

Следование по заданному пути — основное требование для БПЛА. Схема управления с обратной связью для задачи отслеживания пути мобильного робота описана в [38]. Стратегии решения проблемы следования заданной траектории разбиваются на две группы: геометрические и теоретические.

Геометрические методы представляют собой преследование целевой точки [10], законы наведения в пределах прямой видимости (Line-Of-Sight LOS) [18, 29, 33, 37], а также их комбинации [22, 32]. Такие алгоритмы используют виртуальную целевую точку (Virtual Target Point VTP). Законы наведения предписывают БПЛА преследовать VTP, которая в конечном итоге приводит его на заданную траекторию. Расстояние между VTP и положением БПЛА, проецируемым на траекторию, называется виртуальным расстоянием. Стабильность законов наведения LOS существенно зависит от выбора параметра виртуального расстояния [8]. Законы преследования и наведения в прямой видимости могут быть объединены для создания нового закона наведения для следования по пути [22]. Вместо использования преследования или наведения в прямой видимости в [32] был разработан закон нелинейного наведения (Non-Linear Guidance Law NLGL) с использованием VTP. Обычно законы следования по пути выводятся из анализа устойчивости по Ляпунову, который гарантирует глобально устойчивую сходимую к желаемому пути. Реализация данной концепции показана в [43] и [26]. Другой вариант векторного поля предложен в [9], которое называется наведением касательного векторного поля. А также улучшение данного метода, способного обрабатывать медленные изменения ветра в [8]

адаптивный VF.

Теоретические методы управления в частности методы нелинейного управления, популярны для приложений отслеживания пути. Они обеспечивают определенную устойчивость к ветровым возмущениям. Общий подход, используемый при отслеживании пути, основан на управлении Пропорционально-интегрально-дифференцирующим (ПИД) регулятором [41], но не работает так же хорошо, как NLGL [32]. ПИД регулятор с прямой связью, разработанный в [35] работает лучше чем NLGL.

Также несколько типов теоретико-управляющих методов были разработаны для задач следования по пути БПЛА. Некоторые хорошо известные методы — это линейно-квадратичный регулятор (LQR) [1,7,24], управление скользящим режимом [20], управление с прогнозированием модели [42], управление с шагом назад [2,44], теории планирования усиления [11] и динамического программирования [46], а также управление на основе кусочно-аффинных функций [39].

## **2.2. Методы фильтрации ошибок в управляющем сигнале**

Ошибки или неопределенности в управляющем сигнале обычно представляют собой случайный процесс. Задача предсказания такого случайного процесса наиболее типична для калмановской фильтрации, которая базируется на работах Р. Е. Калмана и Р. С. Бьюси [21]. Помимо статистических существуют минимаксные постановки задач. В них предполагается, что неопределенности лишь ограничены в некотором смысле, в остальном они могут быть произвольными. В таких постановках при заранее известном уровне возмущений получают предсказания в виде множеств, размеры которых стабилизируются со временем [19,34]. При этом нет возможности получить обоснованные точные оценки. Дальнейшее практическое использование оценок-множеств ведет к сложным задачам робастной устойчивости [23].

Задача определения ошибки усложняется малым разнообразием вход-



ных данных. Возможность добавить особый сигнал в системе управления может облегчить проблему реконструкции неизвестных параметров. Например, в [49] серия одиночных импульсов во входном канале позволяет восстановить импульсную функцию объекта. Кроме того, специальные рандомизированные сигналы во входном канале дают возможность определить параметры объекта управления, когда рассматривается модель объекта с почти произвольными аддитивными помехами. Процедура, предложенная в [51, 52], работает для любых помех и не требует априорных знаний об их характеристиках. Восстановление неизвестных значений параметров обеспечивается свойствами рандомизированных тестовых сигналов, которые добавляются в контуре управления к собственным сигналам адаптивного управления, поступающим от обратной связи.

### **2.3. Методы измерения ветра на БПЛА с фиксированным крылом**

Для оценки ветра при полете БПЛА с фиксированным крылом применяют несколько методов, которые условно можно разделить на две группы: с использованием специальных датчиков и без них. Одним из методов измерения трехмерного ветра является алгоритм с использованием зонда с несколькими отверстиями (multi-hole-probe algorithm, МНРА), [27, 28], в сочетании с данными об ориентации и положении БПЛА в пространстве, а также его скорости. Достижимое высокое разрешение и точность этого метода требуют точного и быстрого ИНС, а также измерения давления с помощью зондов с несколькими отверстиями. В [12] описан алгоритм трубки Пито (pitot tube algorithm, РТА), который не требует зонда с несколькими отверстиями, а только статической трубки Пито для измерения динамического давления, что делает его менее сложным и менее дорогим. Также скорость ветра можно оценить без датчиков потока воздуха на борту, например, с помощью алгоритма NFSA (no-flow-sensor algorithm), опубликованного в [53]. NFSA использует только информацию о путевой скорости и азимуте траек-

тории полета, получаемые от GNSS. Методы без использования специальных датчиков являются более сложными, но при этом наименее дорогостоящим по сравнению с предыдущими. На рассматриваемом в данной работе БПЛА с фиксированным крылом, использован алгоритм трубки Пито, как компромисс между стоимостью, сложностью и точностью получаемых данных.

## 2.4. Адаптивный рандомизированный алгоритм

В этом разделе представлен краткий обзор основных концепций, связанных с алгоритмом SPSA, более подробные технические детали описаны в соответствующей литературе [40].

Рассмотрим задачу поиска минимума дифференцируемой функции потерь:

$$L_n(\theta) : R^p \rightarrow R,$$

(нижний индекс  $n$  используется для того, чтобы показать, что на функцию потерь влияет шум, распределение которого должно удовлетворять нескольким важным условиям [40])

Существует множество стохастических алгоритмов, которые могут использоваться, чтобы найти такое приближение  $\theta$  (скажем  $\theta^*$ ), при котором  $L_n(\theta)$  минимальна. Метод SPSA вычисляет приближение  $\theta$  на итерации  $k + 1$  следующим образом:

$$\theta_{k+1} = \hat{\theta}_k - a_k \hat{g}_k(\hat{\theta}_k),$$

где  $g_k$  — оцениваемый градиент на  $k$ -й итерации и  $a_k$  — коэффициент, который планируется уменьшать с каждой итерацией

$$a_k = \frac{a}{(k + A)^\alpha},$$

где  $a$ ,  $A$  и  $\alpha$  — заранее выбранные коэффициенты конфигурации. Для оценки  $\hat{g}_k(\cdot)$  используется следующий метод ”одновременных возмущений”. Пусть  $\Delta_k = [\Delta_{k1} \Delta_{k2} \dots \Delta_{kp}] \in R^p$  вектор независимых случайных

величин с нулевым средним (удовлетворяющих условиям, описанным в [40]), и пусть последовательность векторов  $\Delta_k$  — независимая последовательность с  $\Delta_k$  независимыми от  $\hat{\theta}_j, j = 0, 1, \dots, k$ . Базовый метод SPSA (bSPSA) вычисляет две новые точки в пространстве решений и оценивает соответствующие потери следующим образом

$$\begin{aligned} y_k^+ &= \hat{L}_n(\theta_k + c_k \Delta_k) \\ y_k^- &= \hat{L}_n(\theta_k - c_k \Delta_k), \end{aligned}$$

где  $c_k$  — последовательность коэффициентов:

$$c_k = \frac{c}{(k+1)^\gamma},$$

где  $c$  и  $\gamma$  — положительные коэффициенты конфигурации. Далее оценка градиента на  $k$ -й итерации вычисляется по следующей формуле:

$$\hat{g}_k(\hat{\theta}_k) = \frac{y_k^+ - y_k^-}{2c_k} [\Delta_{k1}^{-1} \Delta_{k2}^{-1} \dots \Delta_{kp}^{-1}]^T.$$

Следует отметить, что все элементы вектора  $\theta$  возмущены одновременно, и что только две меры потери необходимы для оценки градиента независимо от размера  $\hat{\theta}$ . На практике последовательность  $\Delta_k$  независимо генерируют по распределению Бернулли, дающему  $\pm 1$  с вероятностью  $\frac{1}{2}$  для каждого значения. При этом возмущения имеют одинаковую амплитуду для всех составляющих  $\theta$ . Было доказано [40], что при определенных условиях  $\hat{g}_k(\cdot)$  как оценка  $g(\cdot)$ , стремится к нулю при  $k \rightarrow \infty$  и  $\theta_k$  сходится "почти наверное" к  $\theta^*$ .

## 2.5. Автопилот PX4

PX4 — модульный многопоточный фреймворк с открытым исходным кодом для глубоко встраиваемых платформ [25]. Он использует многопоточность, чтобы запускать каждый модуль в отдельной задаче либо потоке. Для передачи информации используется обмен сообщениями uORB, в основе которого лежит паттерн издатель-подписчик. PX4 предоставляет Unix-подобные программные интерфейсы для приложе-

ний микроконтроллера. Это позволяет повторно использовать общие знания и опыт Unix подобных систем. Архитектура системы позволяет напрямую взаимодействовать с ROS и запускать отдельные процессы либо как собственные задачи ROS в Linux, либо как задачи на микроконтроллере, обеспечивая максимальную совместимость. Среда выполнения на основе микроконтроллера имеет значительно меньшую задержку и лучшую аппаратную связь, чем типичная система Robotics Linux, и поэтому хорошо подходит для быстрых и высокоскоростных задач управления. PX4 это академическая разработка на базе университета Цюриха. В этой прошивке прозрачная архитектура, предрасположенная к расширению [45].

### 3. Алгоритм оценки ветра

В данной части работы приводится описание разработанного алгоритма для оценки ветра, некоторых его особенностей, а также проблем, которые они решают.

#### 3.1. Система координат

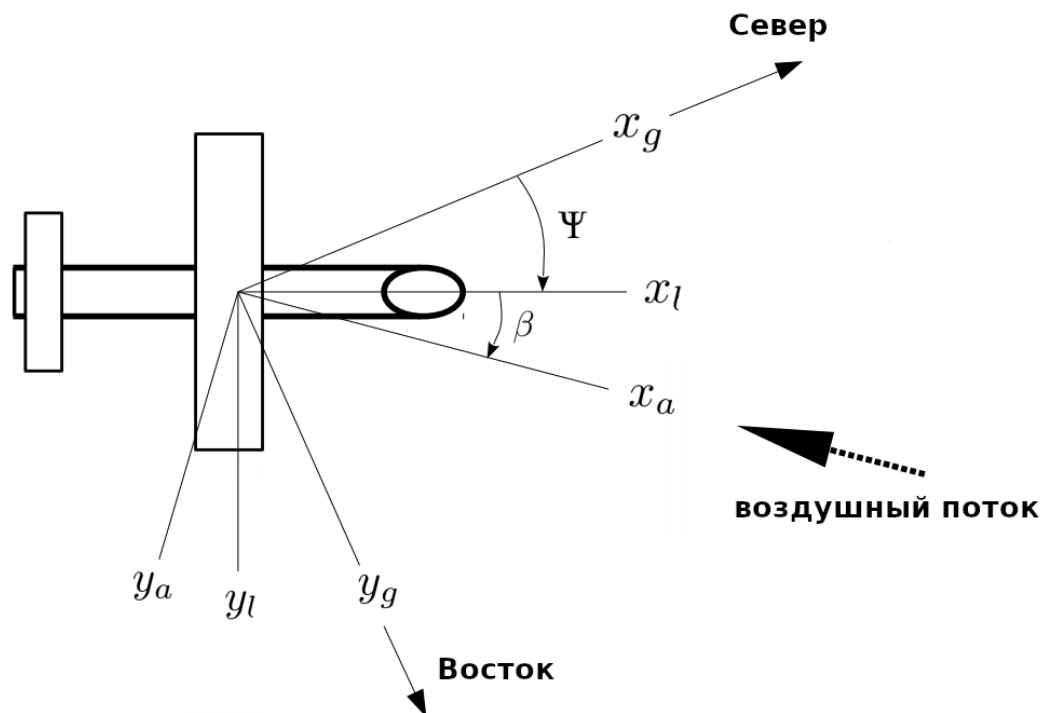


Рис. 1: Вид сверху на измерение ветра с индексами  $a$ ,  $l$  и  $g$ , представляющими, соответственно, скоростную, локальную и глобальную системы координат.  $\Psi$  угол рыскания или истинный курс БПЛА и  $\beta$  — угол бокового скольжения между скоростной системой координат и локальной системой координат.

Для оценки ветра согласно, Бойфферу [13], могут использоваться три декартовых системы координат (см. рис. 1). Первая — это земная система координат или глобальная система координат с индексом  $g$ . В ней ось  $x$  направлена на север, ось  $y$  направлена на восток и ось  $z$  направлена вниз. Вторая — связанная или локальная система координат с индексом  $l$ . При ее использовании начало координат находится в

центре тяжести самолета; ось  $x$  направлена вперед, ось  $y$  направлена в сторону правого борта и ось  $z$  направлена вниз. И третья — скоростная система координат с индексом  $a$ , которая привязана к воздушной скорости БПЛА, то есть его скорости относительно воздуха. Она имеет то же начало, что и локальная система координат, ось  $x$  располагается вдоль воздушного потока, ось  $y$  находится в плоскости симметрии БПЛА и расположена перпендикулярно потоку и ось  $z$  направлена вниз относительно БПЛА.

### 3.2. Вычисление вектора ветра

Вектор ветра  $\vec{w}$  представляет собой направление и силу воздушного потока. Нестационарный наблюдатель, например, летящий БПЛА, видит только относительную скорость  $\vec{u}$ , но с фиксированной точки наблюдения, например, относительно глобальной системы координат, БПЛА движется со скоростью  $\vec{v}$ , которая является суммой  $\vec{u}$  и  $\vec{w}$ . Это соотношение является основой всех методов измерения ветра на БПЛА с фиксированным крылом. Вектор ветра в глобальной системе координат представляет собой разность между векторами  $\vec{v}_g$  и  $\vec{u}_g$ . Вектор скорости БПЛА  $\vec{v}_g$  обычно измеряется с помощью данных GNSS и может быть измерен с достаточно хорошей точностью, тогда как вектор истинной воздушной скорости  $\vec{u}_g$  относительно БПЛА является более сложным параметром для оценивания, как и положение самолета в воздухе, например в углах Эйлера. Согласно Банге [6], вектор ветра может быть вычислен по следующей формуле:

$$\vec{w}_g = \vec{v}_g + \mathbf{T}_{gl}(\vec{u}_l + \vec{\Omega}_l \times \vec{L}), \quad (1)$$

где  $\vec{u}_l$  — истинный вектор воздушной скорости в локальной системе координат БПЛА,  $\mathbf{T}_{gl}$  — матрица преобразования из глобальной в локальную систему координат,  $\vec{\Omega}_l$  — вектор локальных угловых скоростей БПЛА, а  $\vec{L}$  — плечо рычага вектора  $\vec{\Omega}_l$ , представляющее расстояние между ИНС и датчиком измерения воздушной скорости. Существует два основных подхода к измерению вектора истинной воздушной ско-

рости. В первом подходе вектор истинной воздушной скорости БПЛА может быть измерен в скоростной системе координат и затем переведен в глобальные координаты. Во втором подходе вектор истинной воздушной скорости может быть вычислен через изменения вектора  $\vec{v}_g$  в глобальной системе координат при условии, что вектор направления и сила ветра постоянны. Первый подход можно рассматривать как прямое измерение, при котором необходимо измерение вектора относительного ветра, а также данные о текущей позиции и ориентации БПЛА в пространстве. Если эти величины измеряются быстро и точно, небольшие колебания вектора ветра разрешаются во времени и пространстве. Если одна из величин отсутствует, следует сделать допущения, чтобы компенсировать это, а также применить усреднение по всей траектории полета. Метод РТА оценивает вектор ветра, который является усреднением за определенный период времени. Усредненные данные, как правило, не позволяют проводить расчеты порывов в потоке воздуха.

### 3.3. Алгоритм трубки Пито (РТА)

РТА использует данные инерциальной навигационной системы (ИНС). В носовой части самолета устанавливается особая статическая трубка Пито. РТА требует временного усреднения, чтобы компенсировать недостающую информацию о перпендикулярных компонентах вектора воздушной скорости БПЛА. Начиная с уравнения 1 вектор ветра равен разности векторов между путевой скоростью БПЛА и вектором истинной воздушной скорости. Не учитывая прямого измерения воздушную скорость можно аппроксимировать только с помощью трубки Пито. Расчет  $\vec{u}_q$  осуществляется наиболее простым способом с использованием давления торможения принципа Бернулли для несжимаемых потоков. С помощью трубки Пито первая компонента вектора вычисляется как  $u_{ax} = \sqrt{2dp_0/\rho}$ . Остальные компоненты остаются неизвестными [47]:

$$\vec{u}_a = \begin{pmatrix} \sqrt{2dp_0/\rho} \\ u_{ay} \\ u_{az} \end{pmatrix}. \quad (2)$$

Трубка Пито установлена так, что она совмещена с  $x_l$  в локальной системе координат. Необходимо оценить боковую составляющую вектора истинной воздушной скорости  $\vec{u}_a$ , при этом несовпадения между скоростной системой координат и системой координат самолета не могут быть учтены. Вычисление истинной воздушной скорости полета  $\vec{u}_a$  подразумевается вдоль оси  $x_l$  и поэтому преобразованием  $\mathbf{T}_{la}$  можно пренебречь. Используется только преобразование  $\mathbf{T}_{gl}$  из локальной системы координат в глобальную:

$$\vec{w}_g = \vec{v}_g + \mathbf{T}_{gl}\vec{u}_a. \quad (3)$$

Поскольку несовпадение между аэродинамической системой и локальной системой координат БПЛА не может быть учтено, вектор истинной воздушной скорости  $\vec{u}_g$  в выражении 2 указывается в локальных координатах с началом в центре тяжести БПЛА;  $x$  находится вдоль фюзеляжа и с положительным направлением вперед,  $y$  положительно при направлении вправо и  $z$  положительно вверх. РТА может дать точную оценку вектора истинной воздушной скорости только в том случае, если  $\alpha = \beta = 0$ , а это значит, что в общем случае норма  $|\vec{u}_a|$  имеет погрешности. Необходимо отделить известные величины  $\vec{v}_g$  и  $u_{ax}$  от неизвестных  $\vec{w}_g$  вектора ветра и оставшихся компонент вектора воздушной скорости  $u_{ay}$  и  $u_{az}$ . Полученная система уравнений получается переопределенной при присоединении дальнейших измерений, обозначаемых индексом  $i$ . При этом решение вычисляется за определенный временной шаг размера  $M$ . Чтобы иметь возможность отделить известные величины от неизвестных, уравнение 3 записывается в векторном виде  $\vec{v}_g = (v_x, v_y, v_z)$  и  $\vec{w}_g = (w_x, w_y, w_z)$ . Положим, что  $\Phi$  определяет угол поворота вокруг оси  $x_l$ ,  $\Theta$  определяет угол поворота вокруг оси  $y_l$  и  $\Psi$  определяет угол поворота вокруг  $z_l$ , тогда матрицу преобразования  $\mathbf{T}_{gl}$  можно представить



следующим образом:

$$\begin{aligned} \mathbf{T}_{gl} &= \begin{bmatrix} T_{1x} & T_{1y} & T_{1z} \\ T_{2x} & T_{2y} & T_{2z} \\ T_{3x} & T_{3y} & T_{3z} \end{bmatrix} \\ &= \begin{bmatrix} \cos\Theta\cos\Psi & \sin\Phi\sin\Theta\cos\Psi - \cos\Phi\sin\Psi & \cos\Phi\sin\Theta\cos\Psi + \sin\Phi\sin\Psi \\ \cos\Theta\sin\Psi & \sin\Phi\sin\Theta\sin\Psi + \cos\Phi\cos\Psi & \cos\Phi\sin\Theta\sin\Psi + \sin\Phi\cos\Psi \\ -\sin\Theta & \sin\Phi\cos\Theta & \cos\Phi\cos\Theta \end{bmatrix}. \end{aligned} \quad (4)$$

А выражения 1 и 3 запишутся как:

$$\begin{bmatrix} w_x \\ w_y \\ w_z \end{bmatrix} = \begin{bmatrix} v_x \\ v_y \\ v_z \end{bmatrix} + \begin{bmatrix} T_{1x}u_{ax} & T_{1y}u_{ay} & T_{1z}u_{az} \\ T_{2x}u_{ax} & T_{2y}u_{ay} & T_{2z}u_{az} \\ T_{3x}u_{ax} & T_{3y}u_{ay} & T_{3z}u_{az} \end{bmatrix}. \quad (5)$$

Чтобы отделить известное от неизвестного, перепишем выражение 5 следующим образом:

$$\begin{aligned} v_x + T_{1x}u_{ax} &= w_x - T_{1y}u_{ay} - T_{1z}u_{az} \\ v_y + T_{2x}u_{ax} &= w_y - T_{2y}u_{ay} - T_{2z}u_{az} \\ v_z + T_{3x}u_{ax} &= w_z - T_{3y}u_{ay} - T_{3z}u_{az}. \end{aligned} \quad (6)$$

Все известные в получившейся системе уравнений можно обозначить как  $\eta_k$ . Для каждого  $\eta \in \{x, y, z\}$  получится три уравнения следующего вида:

$$\eta_k = v_k + T_{1k}u_{ax} = w_k - T_{2k}u_{ay} - T_{3k}u_{az}. \quad (7)$$

При условии, что  $\vec{w}_g$  постоянен в течение временного шага размера  $M$ , выражения 7 могут быть объединены с линейно независимой системой уравнений. При добавлении новых измерений  $i$  в системе появляются

два неизвестных  $u_{ay}^{(i)}$  и  $u_{az}^{(i)}$

$$\begin{bmatrix} \eta_x^{(1)} \\ \eta_y^{(1)} \\ \eta_z^{(1)} \\ \eta_x^{(2)} \\ \eta_y^{(2)} \\ \eta_z^{(2)} \\ \vdots \\ \eta_x^{(N)} \\ \eta_y^{(N)} \\ \eta_z^{(N)} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & -T_{1y}^{(1)} & -T_{1z}^{(1)} & 0 & 0 & \dots & 0 & 0 \\ 0 & 1 & 0 & -T_{2y}^{(1)} & -T_{2z}^{(1)} & 0 & 0 & \dots & 0 & 0 \\ 0 & 0 & 1 & -T_{3y}^{(1)} & -T_{3z}^{(1)} & 0 & 0 & \dots & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & -T_{1y}^{(2)} & -T_{1z}^{(2)} & \dots & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & -T_{2y}^{(2)} & -T_{2z}^{(2)} & \dots & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & -T_{3y}^{(2)} & -T_{3z}^{(2)} & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & \dots & -T_{1y}^{(M)} & -T_{1z}^{(M)} \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & \dots & -T_{2y}^{(M)} & -T_{2z}^{(M)} \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & \dots & -T_{3y}^{(M)} & -T_{3z}^{(M)} \end{bmatrix} \begin{bmatrix} w_x \\ w_y \\ w_z \\ u_{ay}^{(1)} \\ u_{az}^{(1)} \\ u_{ay}^{(2)} \\ u_{az}^{(2)} \\ \vdots \\ u_{ay}^{(M)} \\ u_{az}^{(M)} \end{bmatrix}. \quad (8)$$

Число неизвестных  $n$  и число уравнений  $m$  связаны соотношением  $n = 3 + \frac{2}{3}m$ , поэтому начиная с шага размером  $M = 3$ , система имеет решение. На практике систему уравнений необходимо явно переопределить, чтобы усреднить мелкомасштабные колебания ветра и получить среднее значение ветра. Уравнение 8 решается численно для  $w_x, w_y$  и  $w_z$  методом наименьших квадратов. Полученный вектор ветра  $\vec{w}_g$  в глобальной системе координат является наилучшей оценкой для  $i$  измерений внутри шага размера  $M$ . Следует отметить, что с помощью РТА нельзя найти вертикальную компоненту ветра  $w_z$  с хорошей точностью. Учитывая что угол тангажа  $\Theta$  обычно мал, то и вертикальная составляющая измерения воздушной скорости ( $T_{3x}u_{ax}$  в выражении 5) будет небольшой. Однако это приведет к высокой неопределенности, если угол тангажа будет большим. Кроме того, длинные периоды усреднения, которые необходимы для РТА, будут сглаживать кратковременные изменения направления и силы ветра.

### 3.4. Добавление рандомизации в управление БПЛА при неизвестных изменениях ветра

Для вычисления рандомизированных поправок ветра рассмотрим его воздействие отдельно в горизонтальной плоскости, где оно выражается в отклонении БПЛА от заданной траектории, и по вертикали, где

из-за его воздействия возникает отклонение от удерживаемой высоты. Горизонтальные компоненты ветра  $w_x$  и  $w_y$  для удобства запишем в полярных координатах  $(\theta, b)$ , где  $\theta$  — направление ветра и  $b$  — его сила. Вертикальную компоненту ветра  $w_z$  обозначим за  $h$ . Данные о местоположении вычисляются с интервалом времени  $\delta$ , т.е. в момент времени  $T_k = T_0 + \delta t$  поступают тройки чисел  $(x_t, y_t, z_t)$ .

Для управления БПЛА, т.е. создания последовательности управляющих сигналов, подающихся на исполнительные механизмы, необходимо по наблюдениям  $(\hat{x}_t, \hat{y}_t, \hat{z}_t)$  оценить неизвестные параметры  $\theta, b$  и  $h$ .

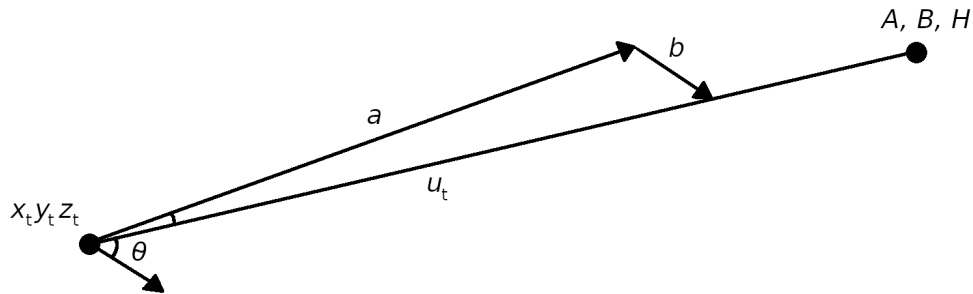


Рис. 2: Направление на цель, курс, смещение от ветра

Пусть задана точка направления на цель  $(A, B, H)$ . В каждый момент времени  $T_t$  БПЛА находится в точке  $(x_t, y_t, z_t)$ . В каждый такт времени от  $T_t$  до  $T_{t+1}$  вычисляется курс, которому должен следовать БПЛА. На этом интервале времени движению объекта в направлении курса  $u$  мешает ветер. Изменения угла направления ветра и его скорости можно оценить по следующим формулам:

$$\theta_{t+1} = \theta_t + \varepsilon_{t+1}$$

$$b_{t+1} = b_t + \xi_{t+1}$$

$$h_{t+1} = h_t + \mu_{t+1},$$

где  $\varepsilon, \xi$  и  $\mu$  — независимые, центрированные и одинаково распределенные случайные величины.

Для оптимизации движения к конечной точке требуется в момент

времени  $T_t$  по последовательности наблюдений  $(\hat{x}_i, \hat{y}_i, \hat{z}_i)_{i=0}^t$ , предложить модель для оценивания  $\theta_{t+1}, b_{t+1}, h_{t+1}$ , минимизирующую среднеквадратичные отклонения:

$$E \left\{ (\theta_{t+1} - \hat{\theta}_{t+1})^2 \right\} \rightarrow \min$$

$$E \left\{ (b_{t+1} - \hat{b}_{t+1})^2 \right\} \rightarrow \min$$

$$E \left\{ (h_{t+1} - \hat{h}_{t+1})^2 \right\} \rightarrow \min.$$

### 3.5. Построение алгоритма оценки ветра

На каждой итерации работы автопилота вычисляется курс  $\psi_t$ , который необходимо удерживать в следующий такт времени, чтобы выйти на заданную траекторию. Также на каждой итерации известно текущее направление БПЛА  $\phi_t$ , а соответственно и ошибка в курсе движения  $\varepsilon_t^\theta$ . также вычисляется расстояние до заданной траектории  $\varepsilon_t^b$  и расстояние до заданной высоты  $\varepsilon_t^h$ :

$$\varepsilon_t^\theta = \psi_t - \phi_t$$

$$\varepsilon_t^b = \text{distance}((x_t, y_t), \text{line}(A_n, B_n, A_{n+1}, B_{n+1}))$$

$$\varepsilon_t^h = \text{abs}(z_t - H).$$

Для оценки параметров ветра строится следующий рандомизированный алгоритм:

1.  $\hat{\theta}_0 = \hat{b}_0 = \hat{h}_0 = 0, \alpha_1, \alpha_2, \alpha_3, \beta > 0$ .
2. Чтение значений переменных полета БПЛА.
3. Генерация последовательности  $\Delta_n$  независимых, одинаково распределенных случайных величин, равных  $\pm\beta$  с одинаковой вероятностью  $\frac{1}{2}$ , называемых пробными рандомизированными возмущениями.
4. Формирование оценки изменения параметров ветра

$$\hat{\theta}_{t+1} = \hat{\theta}_t - \alpha_1 \Delta_{t1} \varepsilon_t^\theta$$

$$\hat{b}_{t+1} = \hat{b}_t - \alpha_2 \Delta_{t2} \varepsilon_t^b$$
$$\hat{h}_{t+1} = \hat{h}_t - \alpha_3 \Delta_{t3} \varepsilon_t^h.$$

5. Публикация полученного прогноза ветра для передачи модулю оценки положения.
6. Переход к шагу 2.

## 4. Особенности реализации модуля оценки ветра

В данной главе приводится описание архитектуры автопилота PX4-Autopilot, из каких частей он состоит, а также особенности реализации модуля предсказания ветра.

### 4.1. Архитектура автопилота

На диаграмме 3 представлена общая архитектура автопилота PX4-Autopilot [30]. Она состоит из двух слоев: промежуточного слоя и слоя высокого уровня. Промежуточный слой предоставляет поддержку автономных роботизированных систем, обеспечивает внутреннюю и внешнюю связь, а также системную интеграцию. Слой высокого уровня — это полетный стек, который отвечает за управление полетом. Алгоритмы управления беспилотными средствами, которые поддерживает PX4-Autopilot, находятся в одной кодовой базе. В целом система имеет реактивный дизайн. Это означает, что вся функциональность разделена на компоненты, которые можно заменить или переиспользовать, общение между компонентами происходит посредством асинхронных сообщений, а также система способна справляться с различной рабочей нагрузкой. Исходный код PX4-Autopilot разделен на модули, которые обозначены на диаграмме 3 моноширинным шрифтом. Стрелки показывают наиболее важные потоки данных между модулями. Данные между модулями передаются через шину сообщений uORB [48], реализующей паттерн издатель-подписчик. Реализация данного паттерна предполагает, что система реактивна, то есть она асинхронна и ее состояние будет обновлено сразу, как будут доступны новые данные, все операции и общение между компонентами параллельны, а также системные компоненты могут получать данные из любого места потокобезопасным способом. Такая архитектура позволяет заменять отдельные модули автопилота даже во время исполнения.

Для интеграции модуля оценки ветра необходимо модифицировать

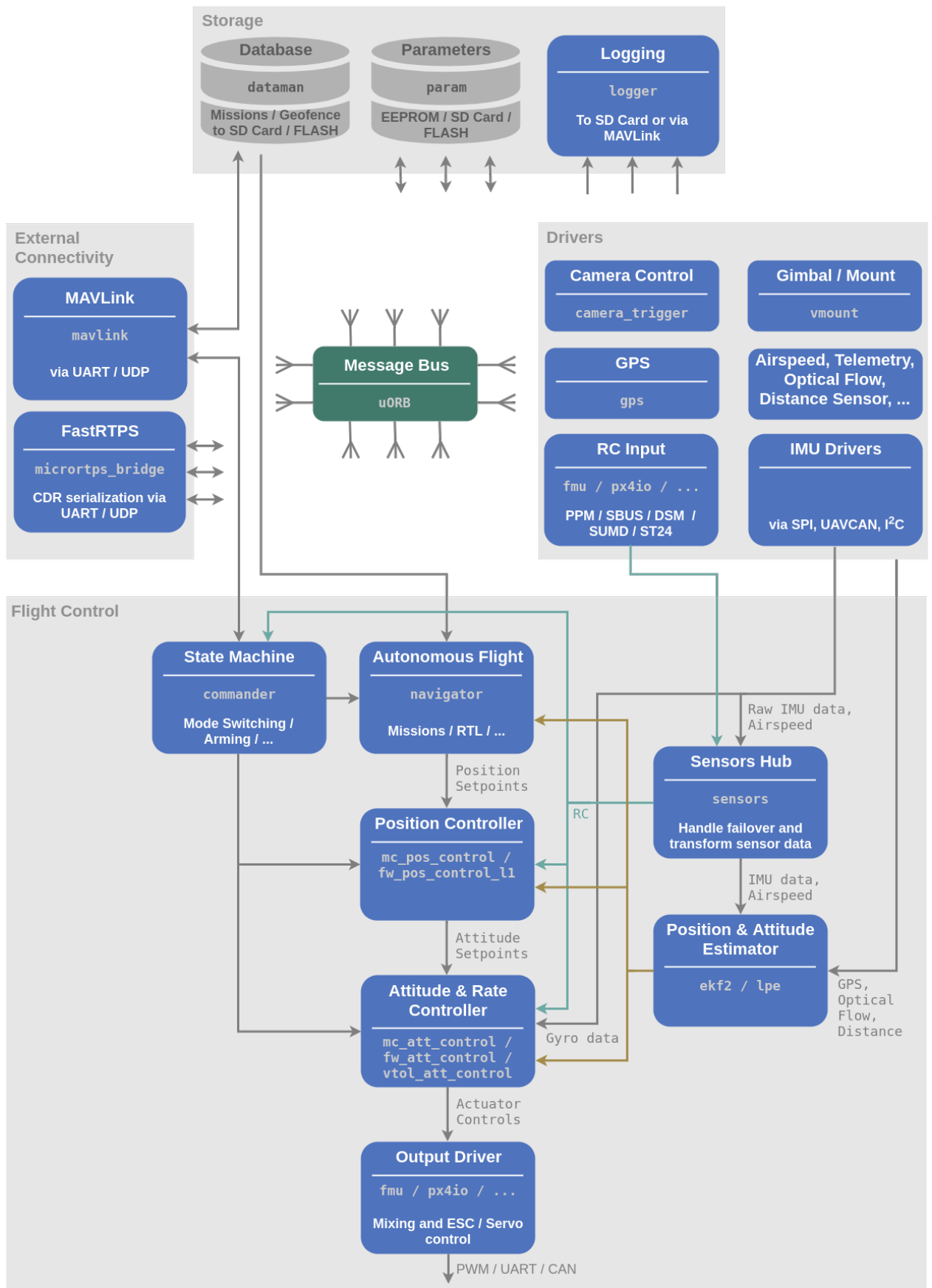


Рис. 3: Архитектура автопилота

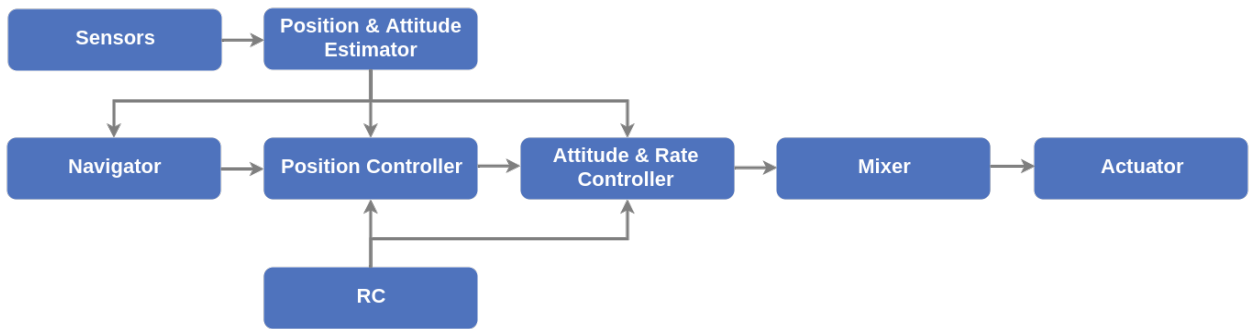


Рис. 4: Компоненты полетного стека

алгоритмы полетного контроллера. Полетный контроллер представляет собой набор алгоритмов для следования заданной траектории, навигации и управления различными беспилотными аппаратами. На диаграмме 4 изображены компоненты, из которых он состоит. Также она показывает полный путь от датчиков, системы навигации и сигналов от пульта управления до моторов и сервоприводов. Компонент оценки принимает на вход данные с одного или нескольких датчиков и на их основе, а также предыдущих шагов, вычисляет состояние аппарата. Компонент управления вычисляет какие команды необходимо задать управляющим элементам, чтобы достичь заданного положения. А компонент миксер сформированные команды для управления системой переводит в конкретные команды для моторов и сервоприводов.

## 4.2. Особенности реализации модуля оценки ветра

В ходе данной работы был реализован модуль оценки ветра для автопилота PX4-Autopilot. Он запускается в общей рабочей очереди с частотой 10 Hz. На каждой итерации происходит чтение следующих сообщений:

- `vehicle_attitude` — данные об ориентации БПЛА в пространстве;
- `vehicle_local_position` — локальное положение БПЛА;
- `vehicle_gps_position` — данные от системы навигации;



- `vehicle_global_position` — положение БПЛА в глобальных координатах;
- `airspeed_validated` — данные о воздушной скорости;
- `position_setpoint_triplet` — предыдущая, текущая и следующая навигационные точки.

Для публикации оценки ветра, полученной в ходе работы данного модуля, было создано сообщение `wind_prediction` и реализована его обработка в модуле оценки положения.

## 5. Тестирование и апробация

В данной части работы описывается сравнение точности следования по маршруту автопилота PX4-Autopilot с использованием полученного модуля предсказания ветра и без него.

### 5.1. Имитационное тестирование

Имитационное моделирование было проведено в симуляторе FlightGear [17], в основе которого лежит Модель динамик полета (FDM Engine) [16]. Он позволяет симулировать полеты БПЛА в различных метеорологических условиях. Одним из настраиваемых параметров погоды является ветер. Данный симулятор позволяет задать силу ветра, частоту его изменения и величину изменения по направлению и силе.

Для тестирования полученного модуля было проведено четыре теста: при постоянном ветре, при слабых изменениях ветра, при значительных изменениях ветра и при сильных изменениях ветра. В каждом из тестов самолет должен был следовать по прямой 5 километров, при этом на каждой итерации модуля управления вычислялось расстояние до заданной траектории в пространстве, т.е. ошибка следования по маршруту. Каждый тест запускался для оригинального автопилота PX4-Autopilot и для модифицированного с использованием модуля предсказания ветра. Перед запуском основных тестов было проведено тестирование автопилота с модулем предсказания ветра, в ходе которого были подобраны коэффициенты для метода SPSA  $\alpha = 0.1$  и  $\beta = 0.01$ , дающие наиболее точное следование заданной траектории. Ниже приведена таблица с численными параметрами ветра для каждого теста:

параметры ветра	без изменений	слабые	значительные	сильные
начальная сила	3 м/с	3 м/с	3 м/с	3 м/с
изменение силы	0 м/с <sup>2</sup>	до 2 м/с <sup>2</sup>	до 5 м/с <sup>2</sup>	до 10 м/с <sup>2</sup>
изменение направления	0 град/с	до 5 град/с	до 15 град/с	до 45 град/с

Сравнение ошибок следования по заданной траектории автопилота с модулем оценки ветра и без него:

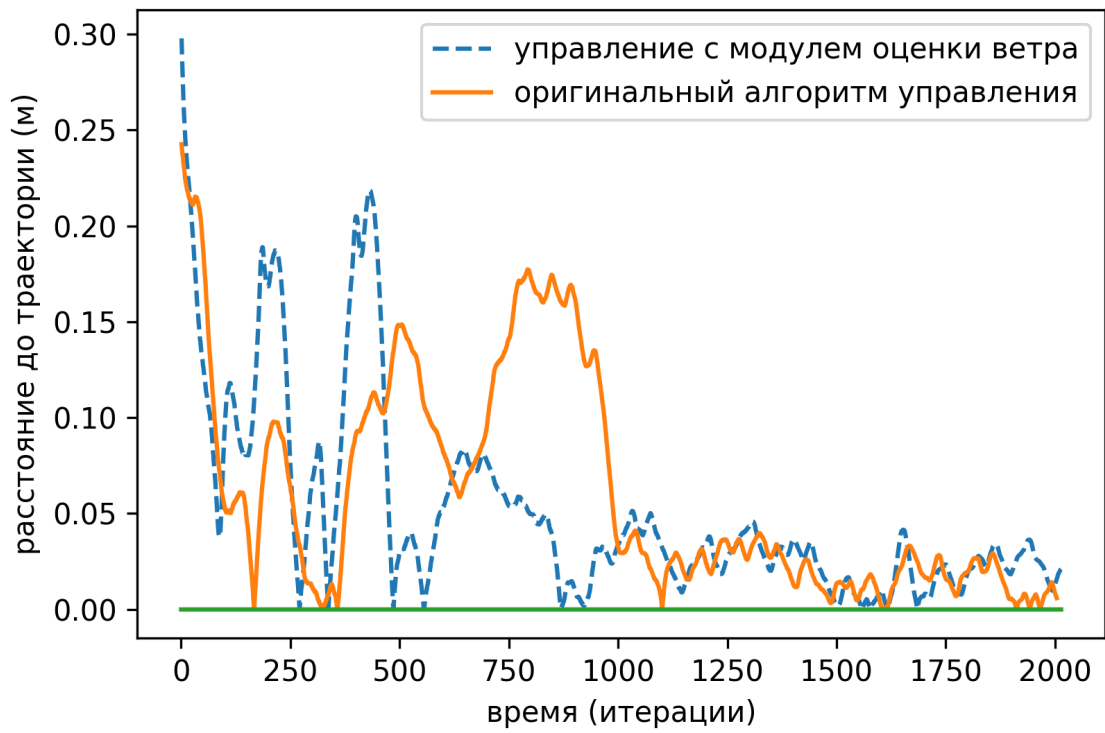


Рис. 5: ошибка при постоянном ветре

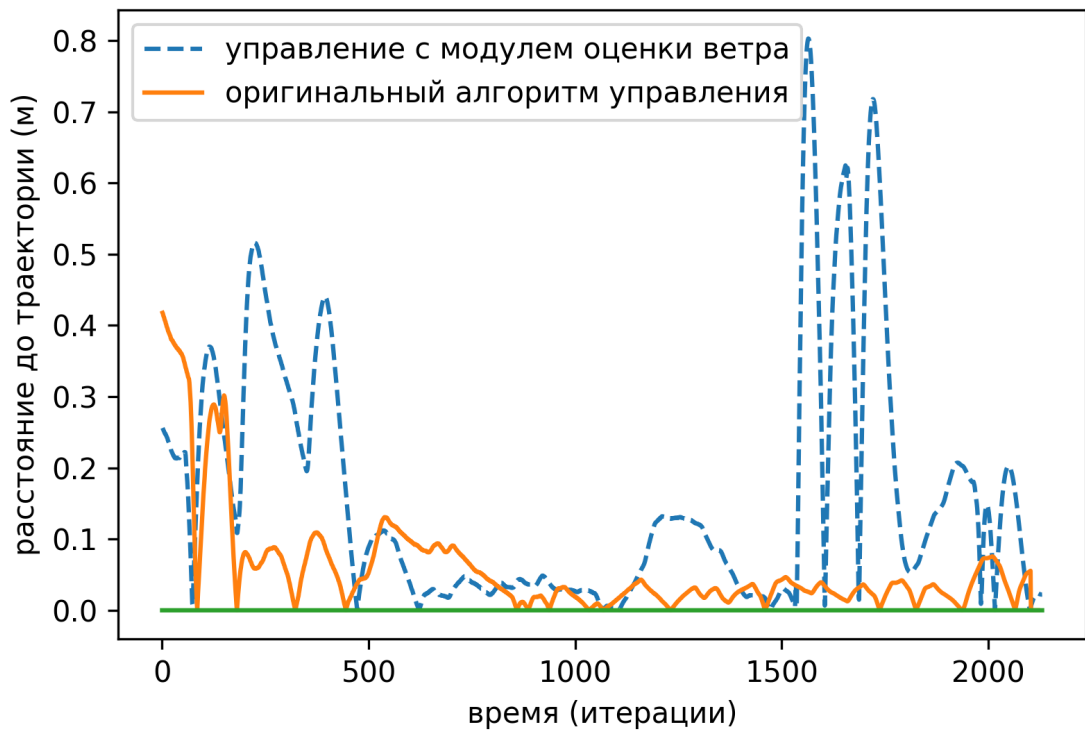


Рис. 6: ошибка при слабых изменениях ветра

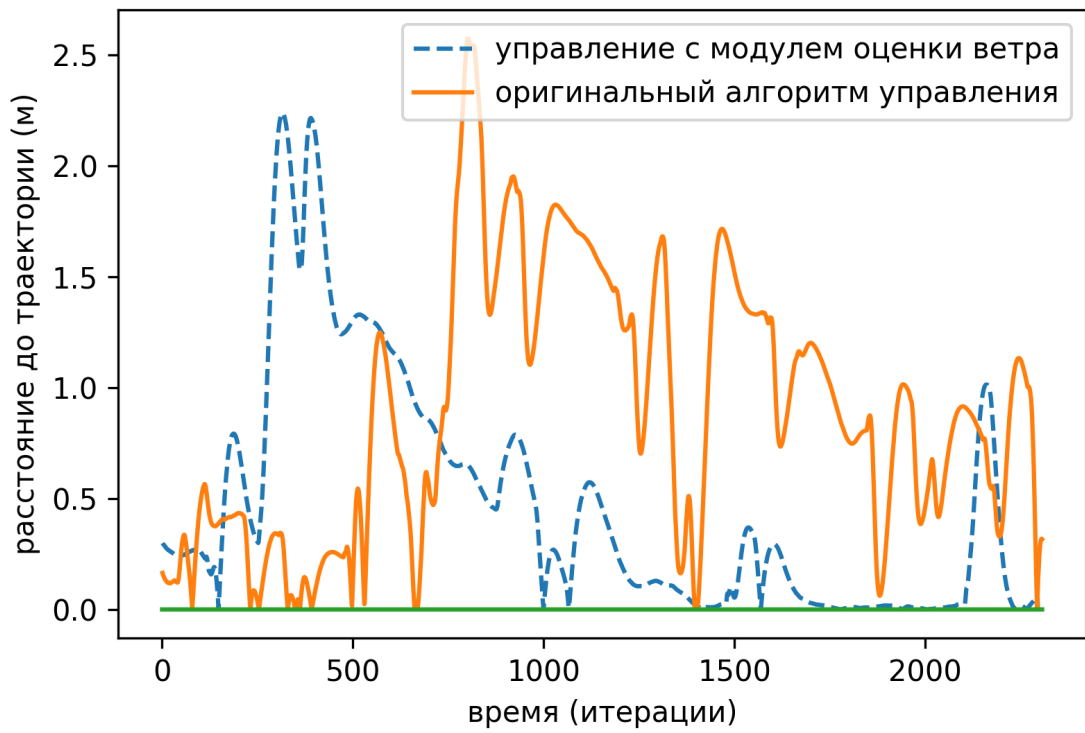


Рис. 7: ошибка при значительных изменениях ветра

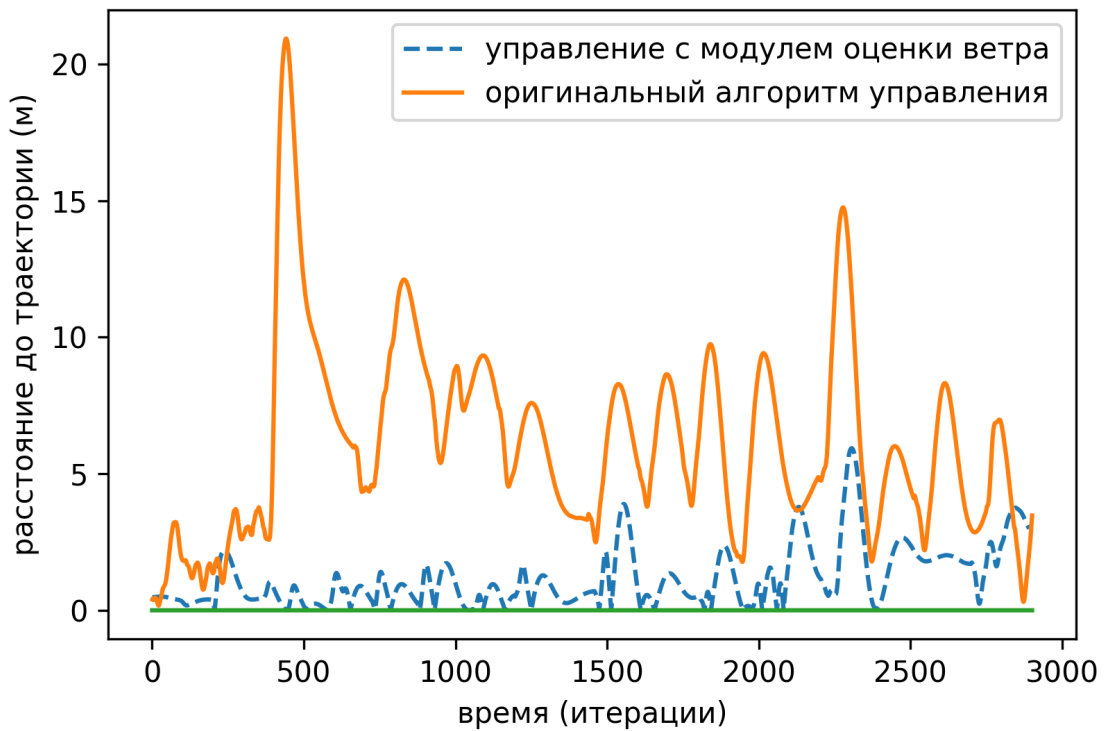
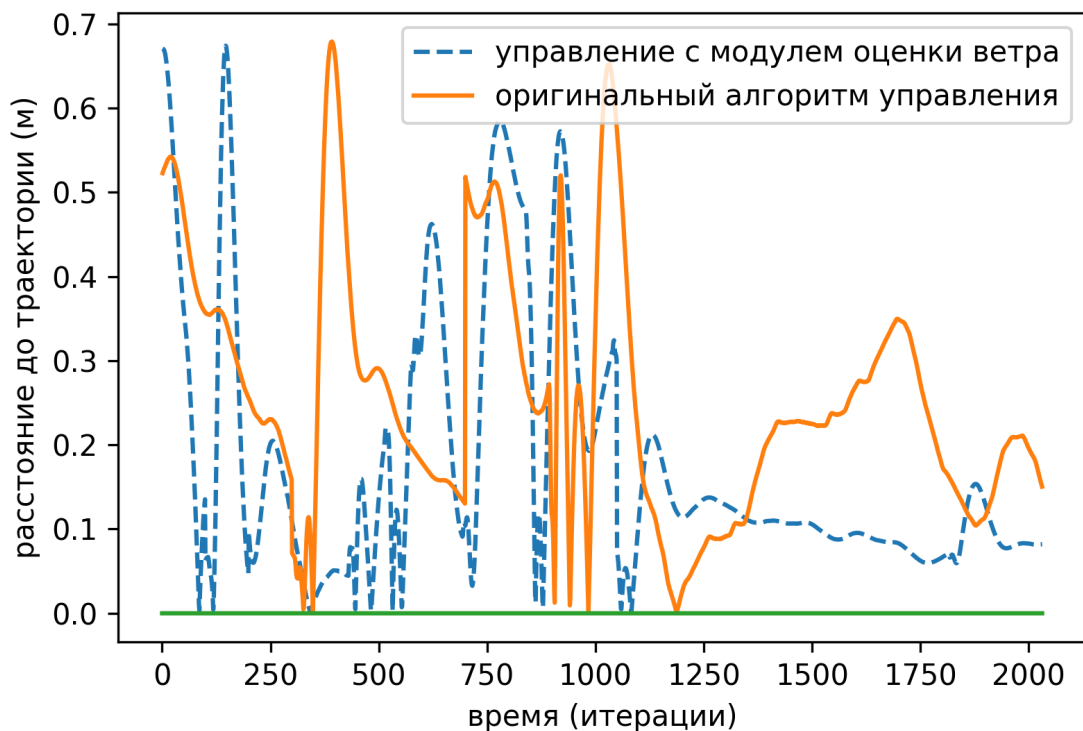


Рис. 8: ошибка при сильных изменениях ветра

Из представленных графиков получается, что при постоянном ветре и незначительных изменениях параметров ветра модуль оценки ветра улучшений не вносит, а иногда из-за того, что намеренно выводит систему из состояния равновесия приводит к большей ошибке — до 1 метра. Однако по мере увеличения изменения силы ветра и его угла направления точность полета с использованием рандомизации возрастает и при сильных изменениях позволяет следовать траектории на несколько метров точнее.

## 5.2. Апробация на реальном БПЛА

Апробация модифицированной прошивки была проведена на БПЛА Диам-20 [50]. Как и при имитационном тестировании измерение ошибки следования заданной траектории проводилось по прямой в 5 километров. Было осуществлено два полета: с использованием модуля оценки ветра и без него. Были получены следующие результаты:



Из данного графика видно, что управление БПЛА с использовани-

ем модуля оценки ветра не уступает управлению без него, а иногда позволяет следовать траектории более точно. Среднее отклонение от траектории при полете без модуля оценки ветра составило 0,28 метра, а с использованием модуля оценки ветра 0,21 метра.

Таким образом, апробация подтвердила результаты имитационного тестирования, и при больших изменениях параметров ветра модуль оценки ветра позволяет улучшить точность следования по маршруту.

## Заключение

В ходе данной работы были получены следующие результаты.

1. Проведен обзор предметной области. Рассмотрены основные подходы к отслеживанию траектории БПЛА и фильтрации ошибок в сигналах, адаптивные алгоритмы управления, алгоритмы с применением линейных фильтров. Был изучен метод SPSA и его применимость относительно поставленной задачи, а также архитектура автопилота PX4-Autopilot и его модуля оценки положения.
2. Создан алгоритм с использованием метода SPSA, который оценивает, какой будет ветер в следующий момент времени. Он состоит из двух частей. В первой вычисляется текущая скорость ветра с помощью алгоритма трубки Пито. Во второй части вычисляется случайное изменение ветра в следующий момент времени с помощью метода SPSA.
3. Спроектирован и реализован модуль для автопилота PX4-Autopilot, который вычисляет текущее значение ветра и генерирует рандомизированные поправки на следующий момент времени по созданному алгоритму, он принимает на вход данные от модуля обработки ошибок и далее передает скорректированные данные в управляющий сигнал автопилота PX4-Autopilot.
4. Проведено тестирование полученной системы на симуляторе FlightGear: предложенный алгоритм при постоянном ветре, за счет добавления пробных возмущение незначительно ухудшает точность следования траектории, однако при значительных изменениях ветра позволяет лететь по заданному маршруту более точно. Также проведена апробация полученной системы на реальном БПЛА Диам-20 самолетного типа, подтвердившая результаты тестирования.

Итак, был создан модуль включения рандомизированных поправок с использованием метода SPSA в управляющий сигнал БПЛА, который

был успешно внедрен в автопилот PX4-Autopilot. Код модифицированного автопилота доступен на GitHub [31].



## Список литературы

- [1] Adaptive Optimal Path Following for High Wind Flights / Ashwini Ratnoo, Sujit P.B, Mangal Kothari, Postdoctoral Fellow. — 2011. — 01.
- [2] Ahmed M., Subbarao K. [Nonlinear 3-D trajectory guidance for unmanned aerial vehicles](#) // 2010 11th International Conference on Control Automation Robotics Vision. — 2010. — P. 1923–1927.
- [3] Alessandri A., Parisini T. Nonlinear modeling of complex large-scale plants using neural networks and stochastic approximation // [IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans](#). — 1997. — Vol. 27, no. 6. — P. 750–757.
- [4] Amelin Konstantin, Granichin Oleg. [Randomized controls for linear plants and confidence regions for parameters under external arbitrary noise](#) // 2012 American Control Conference (ACC). — 2012. — P. 851–856.
- [5] Antal C., Granichin O., Levi S. [Adaptive autonomous soaring of multiple UAVs using Simultaneous Perturbation Stochastic Approximation](#) // 49th IEEE Conference on Decision and Control (CDC). — 2010. — P. 3656–3661.
- [6] Bange J. Coordinated Path Following for Time-Critical Missions of Multiple UAVs via L1 Adaptive Output Feedback Controllers // *Airborne Measurement of Turbulent Energy Exchange between the Earth Surface and the Atmosphere*. — P. 174. — ISBN: [978-3-86844-221-2](#).
- [7] Biao Wang, Xiangxu Dong, Chen B. M. [Cascaded control of 3D path following for an unmanned helicopter](#) // 2010 IEEE Conference on Cybernetics and Intelligent Systems. — 2010. — P. 70–75.
- [8] Bingyu Zhou, Satyavada H., Baldi S. [Adaptive path following for Unmanned Aerial Vehicles in time-varying unknown wind environ-](#)

- ments // 2017 American Control Conference (ACC).— 2017.— P. 1127–1132.
- [9] Chen H., Chang K., Agate C. S. UAV Path Planning with Tangent-plus-Lyapunov Vector Field Guidance and Obstacle Avoidance // [IEEE Transactions on Aerospace and Electronic Systems](#).— 2013.— Vol. 49, no. 2.— P. 840–856.
- [10] Conte Gianpaolo, Duranti Simone, Merz Torsten. Dynamic 3D path following for an autonomous helicopter // [IFAC Proceedings Volumes](#).— 2004.— Vol. 37, no. 8.— P. 472–477.— IFAC/EURON Symposium on Intelligent Autonomous Vehicles, Lisbon, Portugal, 5-7 July 2004. URL: <https://www.sciencedirect.com/science/article/pii/S1474667017320219>.
- [11] Cunha R., Silvestre C., Pascoal A. A path following controller for model-scale helicopters // 2003 European Control Conference (ECC).— 2003.— P. 2248–2253.
- [12] Dobrowolski Bolesław, Kabaciński Mirosław, Pospolita Janusz. A mathematical model of the self-averaging Pitot tube // [Flow Measurement and Instrumentation](#).— 2005.— 08.— Vol. 16.— P. 251–265.
- [13] Done G.T.S. Dynamics of Flight: the Equations. J-L. Boiffier. John Wiley and Sons, Baffins Lane, Chichester, West Sussex P019 1UD, UK 1998. 353 pp. Illustrated. £34.95. // [The Aeronautical Journal \(1968\)](#).— 1999.— Vol. 103, no. 1025.— P. 358–358.
- [14] Duník Jindřich, Straka Ondřej, Šimandl Miroslav. The Development of a Randomised Unscented Kalman Filter // [IFAC Proceedings Volumes](#).— 2011.— Vol. 44, no. 1.— P. 8–13.— 18th IFAC World Congress. URL: <https://www.sciencedirect.com/science/article/pii/S1474667016435780>.
- [15] Experimental validation of Kalman filter-based strain estimation in structures subjected to non-zero mean input / Rajen-

- dra prasath Palanisamy, Soojin Cho, Hyunjun Kim, Sung-Han Sim // [Smart Structures and Systems](#). — 2015. — 02. — Vol. 15. — P. 489–503.
- [16] Flight Dynamics Model. — URL: [https://wiki.flightgear.org/Flight\\_Dynamics\\_Model](https://wiki.flightgear.org/Flight_Dynamics_Model) (online; accessed: 2021-04-21).
- [17] Flightgear flight simulator. — URL: <https://www.flightgear.org/> (online; accessed: 2021-04-21).
- [18] Fossen Thor I., Breivik Morten, Skjetne Roger. Line-of-sight path following of underactuated marine craft // [IFAC Proceedings Volumes](#). — 2003. — Vol. 36, no. 21. — P. 211–216. — 6th IFAC Conference on Manoeuvring and Control of Marine Craft (MCMC 2003), Girona, Spain, 17-19 September, 1997. URL: <https://www.sciencedirect.com/science/article/pii/S1474667017378096>.
- [19] Garulli Andrea, Giarre Laura, Zappa Giovanni. Identification of approximated Hammerstein models in a worst-case setting // [Automatic Control, IEEE Transactions on](#). — 2003. — 01. — Vol. 47. — P. 2046–2050.
- [20] Healey A. J., Lienard D. Multivariable sliding mode control for autonomous diving and steering of unmanned underwater vehicles // [IEEE Journal of Oceanic Engineering](#). — 1993. — Vol. 18, no. 3. — P. 327–339.
- [21] Kim Youngjoo, Bang Hyochoong. [Introduction to Kalman Filter and Its Applications](#). — 2018. — 11. — ISBN: 978-1-83880-536-4.
- [22] Kothari Mangal, Postlethwaite Ian, Gu Da-Wei. A Suboptimal Path Planning Algorithm Using Rapidly-exploring Random Trees // [International Journal of Aerospace Innovations](#). — 2010. — 04. — Vol. 2. — P. 93–104.
- [23] Kuntsevich Vsevolod. Робастная устойчивость и синтез дискретных систем управления нелинейными объектами // [Problemy Ur-](#)

- ravleniya I Informatiki (Avtomatika). — 2007. — 04. — Vol. 39. — P. 5–22.
- [24] Lee Sanghyo, Cho am, Kee Changdon. Integrated waypoint path generation and following of an unmanned aerial vehicle // [Aircraft Engineering and Aerospace Technology](#). — 2010. — 09. — Vol. 82. — P. 296–304.
- [25] Lorenz Meier Dominik Honegger, Pollefeys Marc. PX4: A Node-Based Multithreaded Open Source Robotics Framework for Deeply Embedded Platforms. — 2015. — Accessed: 2020-09-14. URL: <https://people.inf.ethz.ch/pomarc/pubs/MeierICRA15.pdf>.
- [26] [Lyapunov Guidance Vector Fields for Unmanned Aircraft Applications](#) / E. W. Frew, D. A. Lawrence, C. Dixon et al. // 2007 American Control Conference. — 2007. — P. 371–376.
- [27] MASC - a small Remotely Piloted Aircraft (RPA) for wind energy research / Norman Wildmann, M. Hofsäß, F. Weimer et al. // [Advances in Science and Research](#). — 2014. — 05. — Vol. 11. — P. 55–61.
- [28] Measuring the Wind Vector Using the Autonomous Mini Aerial Vehicle M2AV / Aline Van, Den Kroonenberg, Tim Martin et al. // [Journal of Atmospheric and Oceanic Technology - J ATMOS OCEAN TECHNOLOGY](#). — 2008. — 11. — Vol. 25.
- [29] Osborne John, Rysdyk Rolf. Waypoint Guidance for Small UAVs in Wind. — 2005. — 09.
- [30] PX4 Architectural Overview. — URL: <https://dev.px4.io/master/en/concept/architecture.html> (online; accessed: 2020-09-14).
- [31] PX4 Drone Autopilot. — URL: <https://github.com/VladimirMaltsev/PX4-Autopilot> (online; accessed: 2021-04-21).
- [32] Park Sanghyuk, Deyst John, How Jonathan P. Performance and Lyapunov Stability of a Nonlinear Path Following Guidance Method // [Journal of Guidance, Control, and Dynamics](#). — 2007. — Vol. 30, no. 6. — P. 1718–1728. — <https://doi.org/10.2514/1.28957>.

- [33] Path Generation and Tracking in 3-D for UAVs / G. Ambrosino, M. Ariola, U. Ciniglio et al. // [IEEE Transactions on Control Systems Technology](#). — 2009. — Vol. 17, no. 4. — P. 980–988.
- [34] Polyak Boris, Khlebnikov Mikhail. Suppression of bounded exogenous disturbances: Output feedback // [Automation and Remote Control - AUTOMAT REMOTE CONTR-ENGL TR](#). — 2008. — 05. — Vol. 69. — P. 801–818.
- [35] Rhee I., Park S., Ryou C. A tight path following algorithm of an UAS based on PID control // [Proceedings of SICE Annual Conference 2010](#). — 2010. — P. 1270–1273.
- [36] Ribeiro Maria, Ribeiro Isabel. Kalman and Extended Kalman Filters: Concept, Derivation and Properties. — 2004. — 04.
- [37] Rysdyk Rolf. Unmanned Aerial Vehicle Path Following for Target Observation in Wind // [Journal of Guidance, Control, and Dynamics](#). — 2006. — Vol. 29, no. 5. — P. 1092–1100. — <https://doi.org/10.2514/1.19101>.
- [38] Samson Claude. Time-Varying Feedback Stabilization of Car-like Wheeled Mobile Robots // [Int. J. Rob. Res.](#) — 1993. — . — Vol. 12, no. 1. — P. 55–64. — URL: <https://doi.org/10.1177/027836499301200104>.
- [39] Shehab S., Rodrigues L. [Preliminary results on UAV path following using piecewise-affine control](#) // [Proceedings of 2005 IEEE Conference on Control Applications, 2005. CCA 2005](#). — 2005. — P. 358–363.
- [40] [Simultaneous Perturbation Stochastic Approximation](#) // [Introduction to Stochastic Search and Optimization](#). — J. C. Spall, 2003. — P. 176–207. — ISBN: [9780471722137](#). — <https://onlinelibrary.wiley.com/doi/pdf/10.1002/0471722138.ch7>.

- [41] Sun M., Zhu R., Yang X. [UAV Path Generation, Path Following and Gimbal Control](#) // 2008 IEEE International Conference on Networking, Sensing and Control. — 2008. — P. 870–873.
- [42] [Tracking controllers for small UAVs with wind disturbances: Theory and flight results](#) / S. Jackson, J. Tisdale, M. Kamgarpour et al. // 2008 47th IEEE Conference on Decision and Control. — 2008. — P. 564–569.
- [43] Vector Field Path Following for Miniature Air Vehicles / Derek Nelson, Duncan Barber, Tim McLain, Randal Beard // [Robotics, IEEE Transactions on](#). — 2007. — 07. — Vol. 23. — P. 519 – 529.
- [44] [Vision Assisted Autonomous Path Following for Unmanned Aerial Vehicles](#) / V. K. Chitrakaran, D. M. Dawson, H. Kannan, M. Feemster // Proceedings of the 45th IEEE Conference on Decision and Control. — 2006. — P. 63–68.
- [45] Why We Chose PX4 (vs APM) as Luci’s Default Firmware. — URL: <https://medium.com/@Dronesmith/why-we-chose-px4-vs-apm-as-lucis-default-firmware-ea39f4514bef> (online; accessed: 2020-09-14).
- [46] da Silva J. E., de Sousa J. B. [A dynamic programming approach for the motion control of autonomous vehicles](#) // 49th IEEE Conference on Decision and Control (CDC). — 2010. — P. 6660–6665.
- [47] de Jong RianneChor TomásDias Nelson. Medição da velocidade do vento a bordo de um Veículo Aéreo Não Tripulado // Ciência e Natura. — 2011. — URL: <https://www.redalyc.org/articulo.oa?id=467546167015>.
- [48] uORB Messaging. — URL: <https://docs.px4.io/master/en/middleware/uorb.html> (online; accessed: 2021-04-21).

- [49] А. Г. Александров Ю. Ф. Орлов. Конечно-частотная идентификация: динамический алгоритм // Проблемы управления. — 2009. — Р. 2–8.
- [50] Беспилотная авиационная система «ДИАМ 20». — URL: [https://www.diam-aero.ru/bas\\_diam20.html](https://www.diam-aero.ru/bas_diam20.html) (online; accessed: 2021-04-21).
- [51] О.Н. Граничин В.Н. Фомин. Адаптивное управление с использованием пробных сигналов // Автоматика и телемеханика. — 1986. — Р. 100–112.
- [52] О.Н. Граничин Б. Т. Поляк. Рандомизированные алгоритмы оценивания и оптимизации при почти произвольных помехах // Наука. — 2003. — Vol. 293.
- [53] A ‘No-Flow-Sensor’ Wind Estimation Algorithm for Unmanned Aerial Systems / Stephanie Mayer, Gautier Hattenberger, Pascal Brisset et al. // [International Journal of Micro Air Vehicles](#). — 2012. — 03. — Vol. 4. — Р. 15–30.