

Министерство науки и высшего образования Российской Федерации
НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ ТОМСКИЙ
ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ (НИ ТГУ)
Факультет инновационных технологий (ФИТ)
Кафедра информационного обеспечения инновационной деятельности (ИОИД)

ДОПУСТИТЬ К ЗАЩИТЕ В ГЭК
Руководитель ООП,
д.т.н., профессор
В. Сырякин В.И. Сырякин
«20» июня 2019 г.

МАГИСТЕРСКАЯ ДИССЕРТАЦИЯ

ИМИТАЦИОННОЕ МОДЕЛИРОВАНИЕ ДИНАМИКИ ДВИЖЕНИЯ МУЛЬТИКОПТЕРА

по основной образовательной программе подготовки магистров
направление подготовки 09.04.02 – Информационные системы и технологии

Таганов Александр Александрович

Научный руководитель
д.т.н., профессор кафедры УК
С.В. Шидловский С.В. Шидловский
«19» июня 2019 г.

Автор работы
студент группы №18705
А.А. Таганов А.А. Таганов
«19» июня 2019 г.

Томск 2019

Министерство науки и высшего образования Российской Федерации
Национальный исследовательский Томский государственный университет (НИ ТГУ)
Факультет инновационных технологий (ФИТ)
Кафедра информационного обеспечения инновационной деятельности (ИОИД)

УТВЕРЖДАЮ
Руководитель ООП,
д.т.н., профессор
В. Сырякин В.И. Сырякин
«20» июня 2019 г.

ЗАДАНИЕ

по подготовке выпускной квалификационной работы магистра студенту группы
№18705 Таганову Александру Александровичу

1. Тема ВКР: «Имитационное моделирование динамики движения мультикоптера (утверждена приказом по факультету от «09» апреля 2019 г. №30410).

2. Срок сдачи студентом законченной ВКР:

а) на кафедре 17.06.2019 г.

б) в ГЭК 21.06.2019 г.

3. Исходные данные к ВКР: технические средства, используемые в реализации поставленных задач, результаты научно-исследовательской работы, производственных и преддипломной практик.

4. Краткое содержание работы: Изучение основ работы с программами моделирования движения транспортных средств. Рассмотрение теоретических аспектов популярных имитационных моделей динамики движения мультикоптера. Изучение основ работы с программами для расчета аэродинамических и прочностных характеристик мультикоптера. Практическая реализация имитационного моделирования динамики движения мультикоптера. Расчет прочностных и аэродинамических характеристик мультикоптера. Анализ полученных данных, заключение о проделанной работе.

5. Перечень графического материала: информация о предприятии, описание проблем, описание аналогов, постановка цели работы, логическая и физическая модель базы данных, входные/выходные данные, описание интерфейса программы, выводы по работе.

6. Дата выдачи задания 9 апреля 2019 г

Руководитель ВКР,
профессор кафедры УК

Задание принял к исполнению

С.В. Шидловский С.В. Шидловский
А.А. Таганов А.А. Таганов

РЕФЕРАТ

Магистерская диссертация содержит: 83 с., 82 рис., 2 табл., 33 литературных источников.

Список ключевых слов: ИМИТАЦИОННАЯ МОДЕЛЬ, БПЛА, ГЕКСАКОПТЕР, GAZEBO, ROS, ANSYS, ДИНАМИКА ДВИЖЕНИЯ, АЭРОДИНАМИЧЕСКИЕ ХАРАКТЕРИСТИКИ, ПРОЧНОСТНЫЕ ХАРАКТЕРИСТИКИ.

В магистерской диссертации рассмотрены основные принципы и особенности построения имитационной модели динамики движения мультикоптера.

Исследованы различные существующие методы построения имитационных моделей. Изучены основные существующие методы расчета полетных характеристик мультикоптера.

Разработана имитационная модель, целью которой является проверка работоспособности приближенной к реальной модели БПЛА с применением реалистичных сценариев. Приводнён расчет аэродинамических и прочностных характеристик БПЛА.

Для разработки имитационной модели использовался программный пакет Gazebo. Для расчета аэродинамических и прочностных характеристик БПЛА использовался программный пакет ANSYS Workbench 18.2.

Проанализированы результаты работы имитационной модели и результаты расчета характеристик БПЛА.

ABSTRACT

The master's thesis contains: 83 p., 82 fig., 2 tab., 33 literary sources.

The list of keywords: IMITATION MODEL, UAV, HEXACOPTER, GAZEBO, ROS, ANSYS, MOVEMENT DYNAMICS, AERODYNAMIC CHARACTERISTICS, STRENGTH.

In the master's thesis, the basic principles and features of the construction of a simulation model of the dynamics of motion of a multikopter are considered.

Various existing methods for constructing simulation models have been investigated. The main existing methods for calculating the characteristics of the multicopter are studied.

A simulation model has been developed, the purpose of which is to test the performance of a UAV model close to the real one using realistic scenarios. The calculation of aerodynamic and strength characteristics of the UAV is given.

To develop a simulation model, the software package Gazebo was used. To calculate the aerodynamic and strength characteristics of the UAV, the ANSYS Workbench 18.2 software package was used.

Analyzed the results of the simulation model and the results of calculating the characteristics of the UAV.

ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ	6
1 Анализ литературных источников	8
1.1 Выбор модели беспилотного летательного аппарата	8
1.2 Выбор среды моделирования динамики движения мультикоптера	10
1.3 Протокол информационного взаимодействия с дронами	14
1.4 Алгоритм Sense and Avoid	19
1.5 Анализ с технических средств для расчета характеристик БПЛА	21
2 Построение имитационной модели БПЛА и окружающей среды	24
2.1 Описание модели БПЛА и окружающей среды	24
2.2 Имитация работы лидара	33
2.3 Имитация работы камеры	36
3 Основные этапы прочностного и аэродинамического анализа имитационной модели	39
3.1 Твердотельное моделирование	39
3.2 Прочностной анализ имитационной модели	50
3.3 Аэродинамический анализ имитационной модели	57
ЗАКЛЮЧЕНИЕ	78
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ	80

ВВЕДЕНИЕ

На сегодняшний день имитационное моделирование является наиболее эффективным средством исследования и проектирования сложных систем с использованием беспилотных летательных аппаратов.

Актуальность имитационного моделирования повышается с течением времени. Это связано с тем, что исследования над реальным объектом являются либо дорогими, либо невозможными из-за сложности системы. В свою очередь исследования сложных систем с использованием беспилотных летательных аппаратов приводят к некоторым трудностям, моделирование которых позволяет тестировать эти системы еще на этапе их создания, или в момент непосредственной работы с ними, что способствует избеганию неверных результатов при построении данных систем.

На сегодняшний день методы моделирования позволяют создавать имитационные среды, которые предоставляют расширенные возможности при тестировании различного вида систем:

- позволяют создавать упрощенные модели реальных объектов, ориентированные на узкоспециализированных специалистов;
- позволяют создавать детализированные модели и моделируемые системы;
- предоставляют удобные средства калибровки создаваемых моделей, что способствует повышению адекватности модели;
- редактирование и расширение моделей, что позволяет дополнять существующий набор средств разработки моделей.

В магистерской диссертации исследуются популярные программные решения по созданию имитационных моделей. Также рассмотрены вопросы, связанные с решением задачи расчета прочностных и аэродинамических характеристик БПЛА. Рассматриваются процессы работы с некоторыми алгоритмами в имитационной среде и процессы задания реалистичных сценариев для тестирования БПЛА.

Основной целью работы является разработка имитационной модели для изучения динамики движения БПЛА, а также проведение расчетов аэродинамических и прочностных характеристик последнего. Таким образом, объектом исследований является имитационное моделирование. Предметом исследования – существующие имитационные модели для тестирования динамики движения БПЛА мультироторного типа.

Основные задачи:

- 1) Анализ литературных источников, исследование основных требований к созданию имитационных моделей.
- 2) Изучение процессов создания твердотельных моделей в имитационной среде.
- 3) Исследование способов управления твердотельной моделью в имитационной среде.
- 4) Анализ начальных условий для расчета аэродинамических и прочностных характеристик БПЛА.
- 5) Создание работоспособной имитационной модели для тестирования динамики движения БПЛА и расчет аэродинамических и прочностных характеристик БПЛА.
- 6) Анализ результатов работы имитационной модели и полученных расчетов аэродинамических и прочностных характеристик БПЛА.

1 Анализ литературных источников

1.1 Выбор модели беспилотного летательного аппарата

Беспилотные летательные аппараты отличаются от обычных летательных аппаратов отсутствием экипажа на борту. Беспилотные летательные аппараты принято классифицировать по различным признакам, из-за большого количества разновидностей [1].

Базовая классификация БПЛА представлена в таблице 1 [2].

Таблица 1 – Классификация типов БПЛА

	Аэростатические	Аэродинамические			Реактивные
		Гибкое крыло	Фиксированное крыло	Вращающееся крыло	
Безмоторные	Аэростаты	Безмоторные аппараты сверхлегкой авиации	Планеры		
Моторные	Дирижабли	Моторные аппараты сверхлегкой авиации	Летательные аппараты самолетного типа	Летательные аппараты вертолетного типа	Космические реактивные аппараты

На основе представленных данных в таблице 1 для имитационной модели выбран тип БПЛА вертолетного типа. Вертолетные беспилотные летательные аппараты в основном выполняются по двум схемам: вертолетная схема и мультикоптерная схема. В выбранных схемах лежит принцип расположения двигателей и распределению полезной нагрузки. Также необходимо выбрать БПЛА по принципу системы управления.

Существуют три основных принципа для управления беспилотным летательным аппаратом [3]:

- автоматическое управление;
- дистанционно-пилотируемое управление;
- смешанный тип управления.

Отличительной особенностью первого типа управления БПЛА является полная автоматизация выполняемой задачи. Данный тип управления требует высокое качество разработанной системы для решения поставленной задачи, например, следование по траектории на основе датчиков, находящихся на борту БПЛА. Особенностью второго типа управления БПЛА является то, что пилотирование аппарата осуществляется с помощью некоторого передающего устройства в ручном режиме управления. Третий тип управления совмещает особенности первого и второго типа управления. При выполнении задачи в автономном режиме могут измениться условия полета, которые требуют принятия решения от пилота аппарата, в результате чего БПЛА можно переключить на ручной режим управления, избежать критической ситуации и затем вернуться в автоматизированный режим полета [4].

В зависимости от размера рамы выделяют следующие виды БПЛА мультироторного типа [5]:

- класс 250 мм;
- класс 350 мм;
- класс 450 мм;
- класс 550 мм;
- класс 700 мм;
- класс 800 мм.

Числа в названии класса описывают расстояние между осями вращения винтов мультикоптера в миллиметрах. В зависимости от расположения приводов относительно центра масс БПЛА принято классифицировать по следующим схемам [6]:

- трикоптер;
- квадрокоптер;
- гексакоптер;
- октокоптер;
- многороторные мультикоптеры.

Вывод: на основе проведенного анализа существующих разработок БПЛА для создание имитационной модели динамики движения мультикоптера были использованы следующие характеристики:

- для имитационной модели БПЛА был выбран тип БПЛА вертолетного типа, выполненный по схеме мультикоптера;
- по описанным признакам систем управления был выбран автоматический принцип управления БПЛА;
- по размерам рамы был выбран класс 800 мм;
- для описания расположения приводов относительно центра масс выбрана схема БПЛА гексакоптер.

1.2 Выбор среды моделирования динамики движения мультикоптера

Прежде чем приступить к разработке имитационной модели, необходимо сделать выбор инструментария. Он должен быть максимально задокументированным, обладать большим количеством библиотек и расширений, способствующих решению не только задач, поставленных перед разработчиком на данный момент, но и решению задач, которые будут возникать в ходе разработки других модулей беспилотного летательного аппарата [7].

Главным критерием выбора среды моделирования для создания имитационной модели была возможность моделирования поведения беспилотного летательного аппарата в виртуальных 3D сценах и имитирующих вокруг него пространство. В качестве средств визуализации беспилотного летательного аппарата и виртуальных 3D сцен были использованы пакеты Gazebo и Rviz, которые являются частью пакета Robot Operating System [8].

ROS (Robot Operating System) – это мета-операционная система для программирования роботов с открытым исходным кодом. Операционная система предоставляет следующие службы: контроль устройств на борту

исследуемого аппарата, широкий набор функций, передача и получение сообщений между процессами, а также позволяет управлять пакетами. В операционной системе используется архитектура графов, в которой обработка данных происходит в так называемых узлах, которые получают и передают сообщения между собой. Библиотека ориентирована на Unix подобные системы, такие как Ubuntu Linux, Fedora и Mac OS. ROS содержит множество реализаций с открытым исходным кодом общих функциональных возможностей и алгоритмов робототехники. Эти реализации с открытым исходным кодом организованы в «пакеты». Многие пакеты включены в состав дистрибутивов ROS, в то время как другие могут быть разработаны частными лицами и распространены через сайты совместного использования кода, такие как github [9].

ROS предоставляет набор сервисов и библиотек для решения различных задач:

- 1) OpenCV — это библиотека, которая содержит в себе алгоритмы компьютерного зрения и алгоритмы обработки изображений;
- 2) PCL – это библиотека, предоставляющая инструменты для работы с облаками 3D-точек;
- 3) Orocos – это библиотека, позволяющая управлять созданными роботами.

Также в ROS входят драйверы для различных манипуляторов и сенсоров.

Основные преимущества ROS [10]:

- межплатформенная работоспособность: передача сообщений ROS означает, что возможно работать между сильно отличающимися компонентами и подсистемами, которые при этом написаны на разных языках. Это также затрагивает проблему беспорядка API;
- модульность: это связано с распределенной системой сообщений, к примеру, если один компонент выходит из строя, то вся система продолжает работать. Существует огромное количество способов сделать рабочую

систему более надежной, но ROS облегчает для робота продолжение своей работы, даже если не сработали два датчика и мотор у аппарата;

- совместная обработка ресурсов: без ROS чтение и запись ресурсов становятся беспорядочными с большими многопоточными системами. ROS упрощает весь процесс, позволяя потокам не читать и писать на общие ресурсы, а просто публиковать и подписываться на сообщения.

Основной единицей файловой системы является пакет (package). Каждый пакет содержит библиотеки, исполняемые файлы, скрипты, и т. д. Для каждого пакета существует свой манифест (manifest), в котором есть краткое описание пакета, зависимости между пакетами, и различная мета информация, такая как версия пакета, лицензия и т. д. Введем несколько понятий [11]:

- 1) Узел (Node) - некоторый исполняемый процесс, который использует ROS для связи с другими узлами;

- 2) Сообщения (Messages) - специальный тип данных ROS используемый узлом для публикации или подписки на топики;

- 3) Топики (Topics) - узлы могут подписываться на топики, чтобы получать сообщения, или публиковать в топики, чтобы отправлять сообщения;

- 4) Мастер (Master) - имя сервиса для ROS.

- 5) Rosout – эквивалент stdout/stderr.

Gazebo — это программный пакет в ROS, позволяющий моделировать взаимодействие роботов с физическим миром модели. Программный пакет Gazebo предоставляет возможность тестирования работы алгоритмов и возможность физической реализации роботов в моделируемой среде, для выявления недостатков на начальном этапе разработки. Также программный пакет позволяет учитывать некоторые основные взаимодействия робота с реальной средой как трение, скорость полета, имитация сенсоров и датчиков и т.д. Хорошо спроектированный имитатор позволяет быстро тестировать алгоритмы, создавать роботов, выполнять регрессионное тестирование и

тренировать систему ИИ с использованием реалистичных сценариев. Gazebo предлагает возможность точно и эффективно моделировать популяции роботов в сложных помещениях и на открытом воздухе. Программный пакет предоставляет надежный физический движок, высококачественная графика и удобные программные и графические интерфейсы [12].

Большинство графических моделей создаются в программном пакете в SDF формате. SDF — это формат XML, который позволяет описывать объекты и среды для роботов, их визуализацию и типы управления. Данный формат позволяет детализировано описывать конструкции роботов, добавлять статические и динамические объекты, изменять уровень освещения, добавлять изменения в ландшафт и даже физику объекта. В Gazebo есть возможность полностью описать роботизированный объект с использованием SDF формата, независимо от того, является ли робот простым шасси с колесами или гексакоптером. В дополнение к кинематическим и динамическим характеристикам для робота могут быть определены датчики, свойства поверхности, текстуры, трение скольжения и многие другие свойства. Эти функции позволяют использовать SDF для моделирования, визуализации, планирования движения и управления роботом [13].

Программный пакет Gazebo позволяет создавать собственные модели в SDF форматах. Сначала создается модель робота. Она включает в себя описание каждой функциональной части модели: колеса, ноги, винты, шасси, камеры, датчики и т. д. Затем эти части соединяются вместе в отдельный файл SDF-формата и создается модель робота. К функциональным частям робота, которые основаны на передаче некоторых показаний, необходимо использовать плагины. Например, простейший плагин на камеру прикладывает дальность видимости самой камеры к некоторой оси при получении соответствующей команды (например, ROS-сообщения). Существует возможность подключать к моделируемой среде целиком [14].

Rviz (визуализация ROS) представляет собой трехмерный визуализатор для отображения данных датчиков и информации о состоянии от ROS. Трехмерный визуализатор позволяет увидеть текущую конфигурацию в виртуальной модели робота. Трехмерный визуализатор также может отображать живые представления значений датчиков, поступающих на топики ROS, включая данные камеры, измерения инфракрасного расстояния, данные сонара и т. д. Визуализация и регистрация информации о датчиках тестируемой модели является важной частью разработки и отладки виртуальных моделей. Rviz отображает данные 3D-сенсора от стереокамер, лазеров, Kinects и других 3D-устройств в виде точечных облаков или изображений глубины. Данные 2D-датчиков с веб-камер, видеокамер RGB и 2D-лазерных дальномеров можно просматривать в Rviz как данные изображения [15].

Трехмерный визуализатор идеально подходит для выяснения того, что пошло не так в системе зрения. Набор инструментов позволяет мгновенно отобразить или скрыть любую визуальную информацию.

Если фактический робот взаимодействует с рабочей станцией, rviz отобразит текущую конфигурацию робота в модели виртуального робота. Темы ROS будут отображаться в виде живых представлений на основе данных датчиков, публикуемых любыми камерами, инфракрасными датчиками и лазерными сканерами, которые являются частью системы робота. Это может быть полезно для разработки и отладки систем и контроллеров роботов. Rviz предоставляет настраиваемый графический интерфейс пользователя (GUI), чтобы пользователь мог отображать только информацию, относящуюся к данной задаче [16].

1.3 Протокол информационного взаимодействия с дронами

Для разработки имитационной модели БПЛА был использован алгоритм Sense and Avoid. Главная цель данного алгоритма автономная система

избегания столкновения с объектами БПЛА, движущегося по курсу полета. Этот алгоритм работает на полетном контроллере Pixhawk, который поддерживает сообщения MAVLink и прошивку ArduPilot ArduCopter (для микропроцессоров). Чтобы генерировать сообщения MAVLink таким образом, чтобы MAVProху мог интерпретироваться как команды, была потребность в MAVLink интерпретаторе, который использует язык программирования Java в совокупности с алгоритмом Sense and Avoid [17].

На рисунке 1.1 представлен алгоритм построения маршрута БПЛА

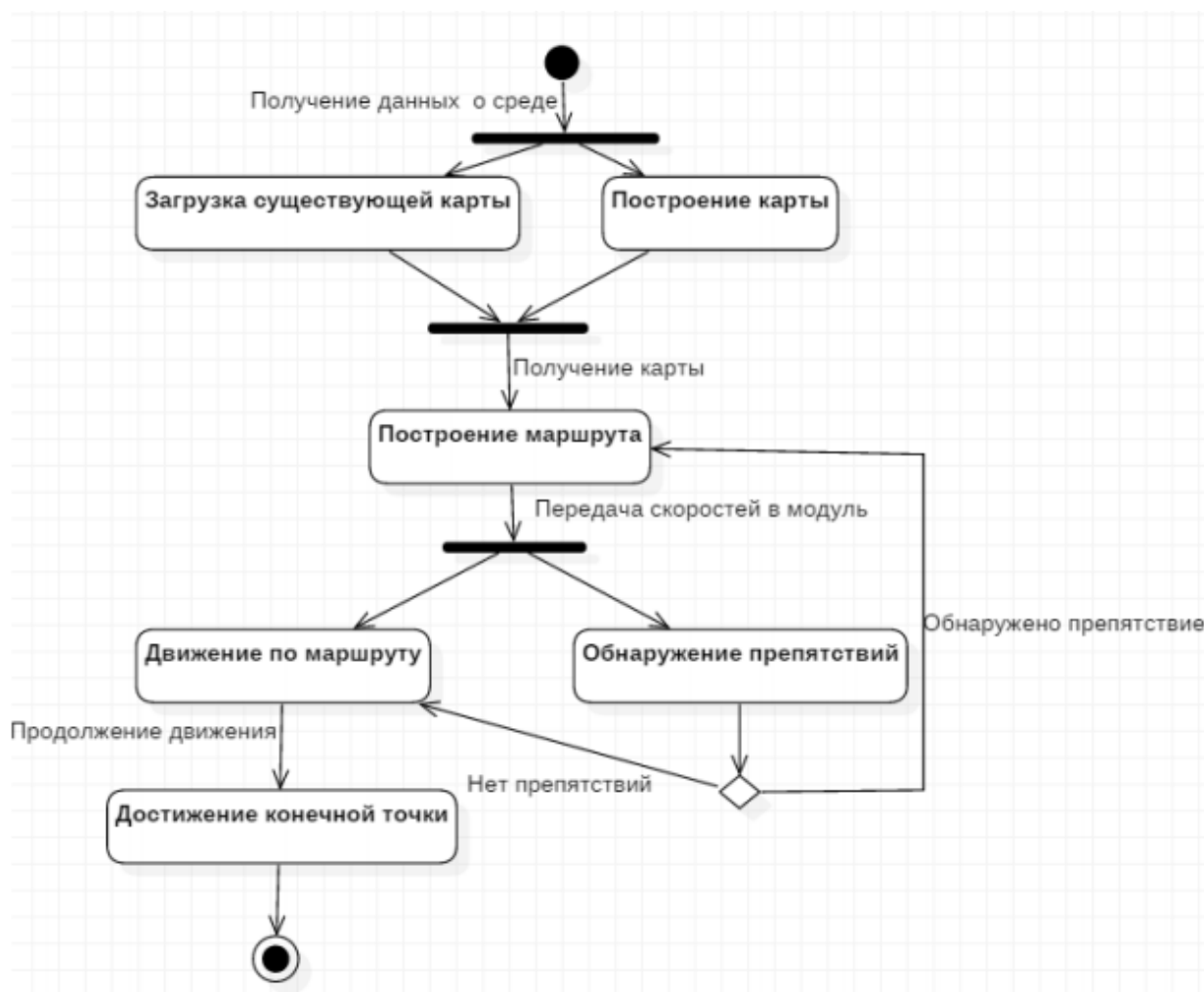


Рисунок 1.1 – Алгоритм построения маршрута БПЛА

Для имитационной модели использовалась библиотека MAVLink. Этот инструмент необходим для чтения сообщения, созданные Pixhawk, и позволяет отправлять команды для MAVLink программно. Этот алгоритм был протестирован на 3D имитируемых средах [18].

MAVLink — это протокол информационного взаимодействия с дронами или малыми беспилотными аппаратами. Протокол описывает информационное взаимодействие между беспилотными аппаратами и наземной станцией управления, а также их составными частями. Базовой сущностью MAVLink является пакет, показанный на рисунке 1.2 [19].

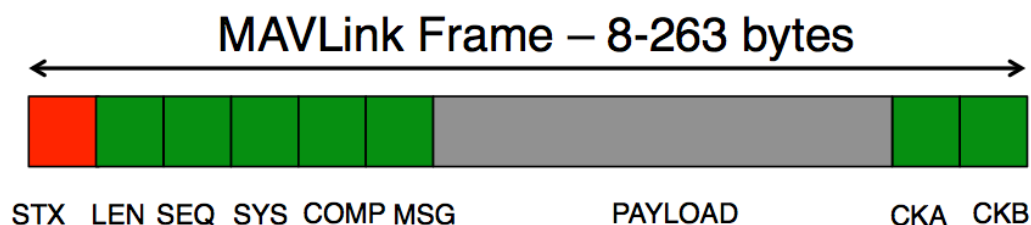


Рисунок 1.2 – Протокол MAVLink

Первый байт пакета (STX) — это символ начала сообщения: 0xFD для версии v2.0, 0xFE для версии v1.0, 0x55 для версии v0.9.

Второй байт пакета (LEN) — это количество полезной нагрузки.

Третий байт пакета (SEQ) — содержит счётчик пакета (0-255), который поможет нам выявить потерю сообщения.

Четвертый байт пакета (SYS) — это идентификатор, отправляющий системы.

Пятый байт пакета (COMP) — это идентификатор отправляющего компонента.

Шестой байт пакета (MSG) — это тип сообщения, от которого зависит какие данные, будут лежать в полезной нагрузке пакета.

Седьмой байт пакета (PAYLOAD) - полезная нагрузка пакета, сообщение, размером от 0 до 255 байт.

Восьмой и девятый байты пакета - СКА и СКВ, нижний и верхний байт, соответственно, содержат контрольную сумму пакета.

Протокол MAVLink может быть использован поверх следующих каналов связи:

- последовательное соединение (UART, USB и др.);

- UDP (Wi-Fi, Ethernet, 3G, LTE);
- TCP (Wi-Fi, Ethernet, 3G, LTE).

MAVLink-сообщения это отдельный пакет данных, передаваемый между задействованными устройствами. Отдельное MAVLink-сообщение содержит информацию о состоянии беспилотного летательного аппарата или команду для беспилотного летательного аппарата [20].

Для обмена данными между различными частями системы (аппарат, сенсоры, автопилот и т. д.) используется операционная система для роботов ROS, в основе которой лежит паттерн publisher-subscriber. Для информационного взаимодействия с аппаратом используется ROS пакет - mavros. Этот пакет реализует возможность посылать команды управления аппаратом по протоколу MAVLINK, используя стандартные ROS типы данных. MAVROS - MAVLink расширяемый узел связи для ROS с прокси для наземной станции управления [21].

Далее будут рассмотрены основы управления Offboard через MAVROS применительно к модели квадрокоптера в Gazebo с запуском SITL.

В то время как MAVROS может использоваться для связи с любым автопилотом с поддержкой MAVLink, пример приведен в контексте обеспечения связи между стеком полета PX4 и совместимым компьютером с поддержкой ROS. ROS может использоваться с PX4 и симулятором Gazebo. Он использует узел MAVROS MAVLink для связи с PX4 [22].

Интеграция ROS и Gazebo с PX4 следует шаблону на диаграмме ниже. Это показывает общую среду моделирования PX4. PX4 связывается с симулятором Gazebo для приема данных датчиков из имитируемого мира и отправки значений двигателей и исполнительных механизмов. Он связывается с GCS и API Offboard для отправки телеметрии из моделируемой среды и получения команд. На рисунке 1.3 представлена интеграция ROS и Gazebo с PX4 [23].

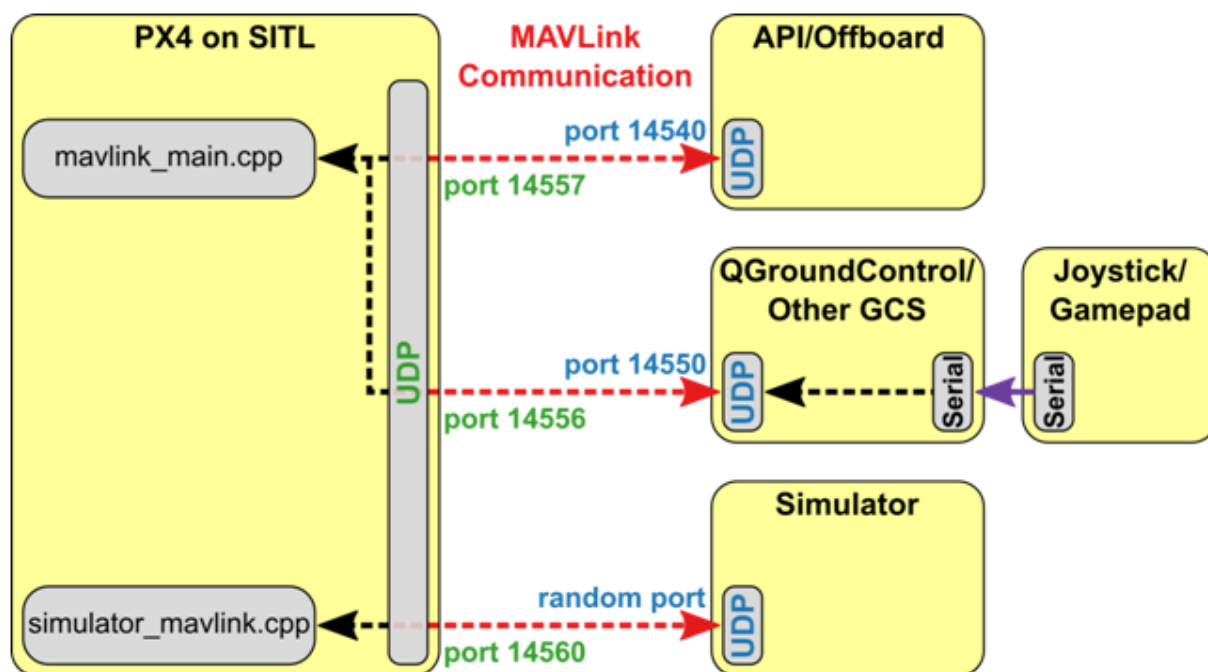


Рисунок 1.3 – Интеграция ROS и Gazebo с PX4

Моделирование Gazebo может быть изменено для интеграции датчиков, публикуемых непосредственно в темах ROS, например, лазерный плагин Gazebo ROS. Чтобы поддержать эту функцию, Gazebo должен быть запущен с соответствующими обертками ROS.

Интерфейс Mavros ROS-MAVLink имеет стандартные реализации для отправки этих сообщений. Они также могут быть отправлены с использованием чистого кода C / C++ и непосредственного использования библиотеки MAVLink.

Основными сервисами MAVROS [24]:

- /mavros/set_mode — установить полетный режим контроллера. Обычно устанавливается режим OFFBOARD (для управления с Raspberry Pi);
- /mavros/cmd/arming — включить или выключить моторы беспилотного летательного аппарата (изменить armed-статус).

1.4 Алгоритм Sense and Avoid

Алгоритм Sense and Avoid опирается на интерфейс LiDAR. Он считывает показания датчика для расчета положений объектов в ходе столкновения, а затем принимает решение для корректировки заданного курса. Алгоритм «Sense and Avoid» начинает работу с получения показаний с LiDAR от PX4 через MAVProxy. Затем запускается циклический (поточный) код. В каждом цикле алгоритма Sense and Avoid производится проверка показаний расстояния LiDAR (автоматически выводит расстояние до объекта в метрах). На рисунке 1.4 представлены функции алгоритма Sense and Avoid [25].

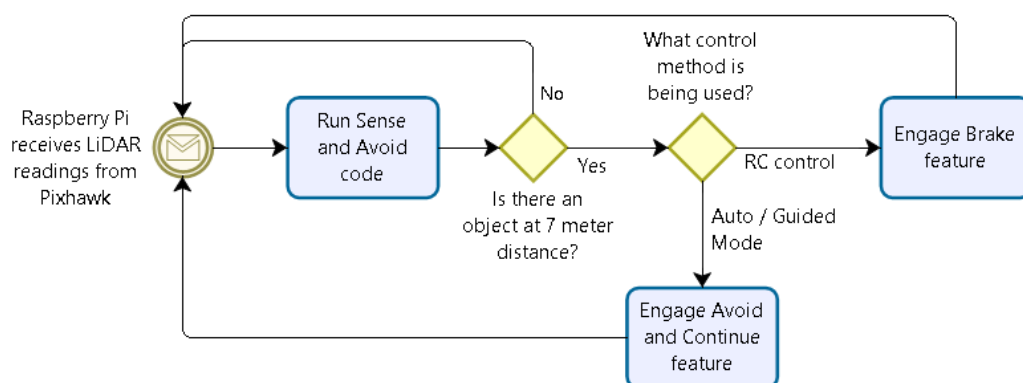


Рисунок 1.4 – Функции алгоритма Sense and Avoid

Если LiDAR не обнаруживает объект, до которого расстояние меньше или равно 7 метрам, это означает, что никаких действий не требуется, и БПЛА не меняет заданный курс полета. Поэтому завершается текущий процесс цикла и запускается новый. Если LiDAR обнаруживает объект, до которого расстояние меньше или равно 7 метрам, алгоритм определяет, какой режим управления активен в данный момент: автономные методы, такие как «Авто» или «Ведомый» режимы полета. Если БПЛА летит в режимах «Авто» или «Ведомый», алгоритм активирует функцию «Avoid and Continue». При активации функции технологический цикл заканчивается. Предыдущий метод управления возобновляется, и новый цикл начинается, как это видно на блок-схеме.

Функция «Avoid and Continue» необходима для полностью автономных задач. Реализация этого алгоритма не только гарантируют предотвращение столкновений объектов, но и возобновляет прежние установленные цели. Блок-схема, представленная на рисунке 1.5, описывает, что представляет из себя функция «Avoid and Continue».

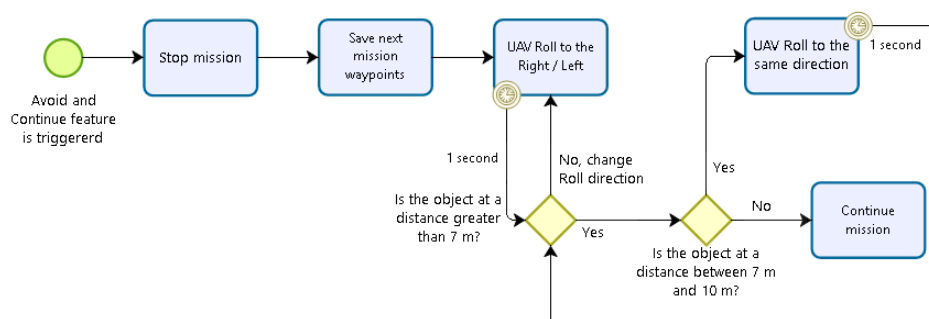


Рисунок 1.5 - Блок-схема «Avoid and Continue»

Миссии БПЛА выполняются автономно, начиная упорядоченно с первой путевой точки до последней точки. Миссия может иметь несколько путевых точек. Если между этими путевыми точками есть препятствие, алгоритм будет запущен. Изменение направления БПЛА при уклонении от препятствия определяется случайным образом. Это связано с тем, что LiDAR имеет узкий луч (расхождение пучка SF11 / C составляет $0,2^\circ$). Поэтому невозможно узнать ширину объекта и выбрать направление движения БПЛА для быстрого избегания столкновения. Но, низкий пучок дивергенции может быть полезным, поскольку он делает измерение расстояния до объекта более точным. Если объект столкновения находится на расстоянии более 10 м, то алгоритм возобновит выполнение миссии, загрузив предыдущие сохраненные путевые точки. Если расстояние между 7 и 10 метрами, то БПЛА будет повторять то же самое, только загрузка предыдущих сохраненных путевых точек составит 2 секунды. Если расстояние ниже 7 метров, то БПЛА отлетит назад на безопасное расстояние 7 метров и продолжит выполнение загруженной миссии.

1.5 Анализ с технических средств для расчета характеристик БПЛА

Достоинством виртуального моделирования является возможность определения слабых мест в конструкциях тестируемых объектов и определения аэродинамических характеристик тестируемых объектов без использования реальных моделей. Методика компьютерного моделирования требует особых навыков работы в специализированных пакетах прикладных программ [26].

Такие программы как Autodesk Inventor, Meshlab и SolidWorks являются доступными техническими средствами для осуществления поставленных задач.

Autodesk Inventor – система трёхмерного твердотельного и поверхностного параметрического проектирования, предназначенная для создания цифровых прототипов промышленных изделий. Инструменты Inventor обеспечивают полный цикл проектирования и создания конструкторской документации. Основные функции Autodesk Inventor:

- 1) 2D и 3D-моделирование;
- 2) создание изделий и получение разверток;
- 3) разработка электрических и трубопроводных систем;
- 4) проектирование оснастки для литья пластмассовых изделий;
- 5) динамическое моделирование;
- 6) параметрический расчет напряженно-деформированного состояния деталей и сборок.

Meshlab – это специализированное программное обеспечение для обработки 3D-сетки. Пакет Meshlab в первую очередь ориентирован на управление большими неструктурированными сетками. Пакет Meshlab предоставляет набор инструментов для их редактирования, очистки, починки, проверки, визуализация и преобразования. Meshlab является бесплатным программным обеспечением с открытым исходным кодом. Это

программное обеспечение является незаменимым для редактирования компьютерных моделей сложной геометрии.

SolidWorks – программный комплекс САПР для автоматизации работ промышленного предприятия на этапах конструкторской и технологической подготовки производства. SolidWorks представляет собой мощный инструмент для 3D моделирования и автоматизированного проектирования сложных изделий различного назначения. Это полноценный набор для конструирования изделий в цифровом виде, который содержит в себе множество дополнительных инструментов, позволяющих производить над моделью виртуальные технические испытания.

Предлагаемые фирмой ANSYS Inc. средства численного моделирования и анализа совместимы с пакетами, в частности система ANSYS сопрягается с CAD-системами NX, CATIA, Pro/ENGINEER, SolidEdge, SolidWorks, Autodesk Inventor и некоторыми другими. На рисунке 1.6 представлена совместимость распространенных САПР на основе технологий Unite с программным пакетом Ansys Workbench 18.2.

Формат	Импорт	Открытие	Обновление	Сохранение
CATIA V4	✓	✓	*	*
CATIA V5	✓	✓	*	*
Siemens NX / Unigraphics® NX	✓	✓	*	*
SolidWorks	✓	✓	*	*
JT	*			*
CADD5® 5	✓		✓	✓
PTC Creo Elements/ Direct®	✓		✓	
STEP	✓			✓
Autodesk Inventor™	✓			
Rhinoceros®	✓			
SolidEdge	✓			
AutoCAD® DXF™	✓			✓
AutoCAD Drawing	✓			✓

Рисунок 1.6 - Совместимость распространенных САПР на основе технологий Unite с программным пакетом Ansys Workbench 18.2

На основе анализа существующих САПР для решения поставленных задач был выбран САПР SolidWorks.

SolidWorks считается неотъемлемой частью промышленных предприятий, задачей которых является разработка и производство изделий различного назначения. Сюда входят инженерные конструкции любой сложности, разнообразные детали и компоненты полноценных систем, и даже электрические схемы. Также нередко встречается применение программы в промышленном дизайне. Данная программа доступна для работы лишь на ОС семейства Windows. Она выбрана для работы не только наличием широкого функционала, а также за счет простоты в освоении и доступности интерфейса.

Ниже приведен базовый функционал программы, доступный без установки расширений:

- 1) твердотельное 3D моделирование;
- 2) разработку сварных конструкций;
- 3) расчеты на прочность;
- 4) просчет гидро/аэродинамики;
- 5) возможность создания чертежей;
- 6) проектирование с учетом материала изделия;
- 7) визуализацию;
- 8) просчет на изгиб;
- 9) работу с данными 3D сканирования (функция ScanTo3D);
- 10) возможность проектирования изделий из листового металла;
- 11) работу с электросхемами;
- 12) возможность анимации готового изделия;
- 13) экспорт данных в различные форматы.

2 Построение имитационной модели БПЛА и окружающей среды

2.1 Описание модели БПЛА и окружающей среды

Необходимая физическая 3D модель робота в Gazebo описывается с помощью файла формата SDF, который обычно используется в ROS.

Рассмотрим файл SDF формата на примере описания модели гексакоптера в Gazebo. Создание модели проходит с помощью тэгов. Описание самой модели прописывается между тэгами. На рисунке 2.1 представлено описание тэга link.

```
<link name='chassis'>
  <pose>0 0 0.1 0 0 0</pose>
  <collision name='collision'>
    <geometry>
      <box>
        <size>0.5 0.2 0.1</size>
      </box>
    </geometry>
  </collision>

  <visual name='visual'>
    <geometry>
      <box>
        <size>0.5 0.2 0.1</size>
      </box>
    </geometry>
    <material>
      <script>
        <name>Gazebo/Orange</name>
        <uri>__default__</uri>
      </script>
    </material>
  </visual>

  <inertial>
    <mass>5</mass>
    <inertia>
      <ixx>0.2</ixx>
      <iyy>0.3</iyy>
      <izz>0.2</izz>
      <ixy>0</ixy>
      <ixz>0</ixz>
      <iyz>0</iyz>
    </inertia>
  </inertial>
</link>
```

Рисунок 2.1 – Описание тэгов link

Ниже приведен список основных тэгов:

- 1) Тэг link определяет базовый структурный элемент, их может быть несколько в одной модели. Внутри link нужно определить тэги:

- pose в формате хуз гру для определения позиции относительно начала координат модели, этот тэг используется очень часто и определяет позицию относительно родительского тэга;

- collision (их может быть большое количество внутри тэга link) используется для просчета столкновений, в нем можно использовать как стандартные фигуры, так и импортированные модели форматов .dae или .stl. Здесь же описываются свойства поверхности вроде трения;

- visual добавляется для обрисовки объектов, здесь также можно использовать как стандартные фигуры, так и импортировать модель форматов .dae или .stl. Здесь же добавляются текстуры;

- inertial описывает физические инерциальные свойства тэга link: его массу, тензор инерции и т.д.

Тэги link, collision, visual обязательно должны быть снабжены уникальными именами. На рисунке 2.2 представлено описание тэгов joint.

```
<joint name="$lr_wheel_hinge" type="revolute">
  <parent>chassis</parent>
  <child>$lr_wheel</child>
  <pose>0 0 ${wheelWidth/2} 0 0 0</pose>
  <axis>
    <xyz>0 1 0</xyz>
  <dynamics>
    <damping>${wheelDamping}</damping>
  </dynamics>
  <limit>
    <effort>100</effort>
    <velocity>100</velocity>
  </limit>
</axis>
</joint>
</macro>
```

Рисунок 2.2 – Описание тэгов joint

2) Тэг joint используется, чтобы соединять тэги link между собой. Ему также нужно задать уникальное имя и тип. Внутри joint нужно определить тэги:

- хуз определяет, по какой оси будет вращение;
- limit задает ограничение на сочленение;
- effort описывает максимальные моменты сил Н*м которые можно задать по оси;

- velocity описывает максимальную скорость;
- upper и lower определяют максимальный и минимальный угол поворота;
- damping определяет возникающее сопротивление пропорционально скорости.

Варьированием этого параметра и момента силы, которая будет прикладываться по оси, задается максимальная скорость вращения сочленения и его динамика. На рисунке 2.3 представлено описание тэгов include.

```
<?xml version="1.0"?>
<sdf version="1.4">
  <world name="mobot">
    <include>
      <uri>model://ground_plane</uri>
    </include>
    <include>
      <uri>model://sun</uri>
    </include>
  </world>
</sdf>
```

Рисунок 2.3 - Описание тэгов include

3) Тэг include позволяет загружать созданные модели в программный пакет Gazebo. Внутри тэгов include можно так же добавить тэг pose, чтобы задать положение, сюда можно добавить все необходимые объекты, чтобы каждый раз не добавлять их вручную. На рисунке 2.4 представлено описание тэгов для камер.

```
<sensor name='camera1' type='camera'>
  <update_rate>30</update_rate>
  <camera name='head'>
    <horizontal_fov>1.39626</horizontal_fov>
    <image>
      <width>800</width>
      <height>800</height>
      <format>R8G8B8</format>
    </image>
    <clip>
      <near>0.02</near>
      <far>300</far>
    </clip>
    <noise>
      <type>gaussian</type>
      <mean>0</mean>
      <stddev>0.007</stddev>
    </noise>
  </camera>
```

Рисунок 2.4 – Описание тэгов для камер

4) Тэги, которые нужны для создания камер:

- horizontal_fov задает широкоугольность камеры;
- clip - область обрисовки, то есть камера видит объекты от near до far;
- noise - добавляет шум.

База данных модели должна подчиняться определенной директории и файловой структуре. В каждом каталоге любых моделей имеется файл формата .config, содержащий метаданные о модели. Конфигурационный файл является обязательным. Каталог models также содержит SDF для модели и любых материалов, сеток и плагинов. Каталог meshes содержит все файлы COLLADA и / или STL для модели.

Папка catkin_ws. В этой папке хранится пакет mavros_offboard, описанный выше, в котором реализована функция отправки скорости, ускорения, а также скорости вращения по курсу на аппарат по протоколу MAVLINK.

Catkin_ws - это папка, в которой производятся изменения, сборка и установка пакетов catkin. Рабочее пространство catkin_ws содержит различные пространства, которые необходимы в процессе разработки программного обеспечения для системы. На рисунке 2.5 представлена директория catkin_ws.

```
workspace_folder/      -- WORKSPACE
src/                   -- SOURCE SPACE
  CMakeLists.txt       -- The 'toplevel' CMake file
  package_1/
    CMakeLists.txt
    package.xml
    ...
  package_n/
    CATKIN_IGNORE     -- Optional empty file to exclude package_n from being processed
    CMakeLists.txt
    package.xml
    ...
build/                 -- BUILD SPACE
  CATKIN_IGNORE       -- Keeps catkin from walking this directory
devel/                 -- DEVELOPMENT SPACE (set by CATKIN_DEVEL_PREFIX)
  bin/
  etc/
  include/
  lib/
  share/
  .catkin
  env.bash
  setup.bash
  setup.sh
  ...
install/               -- INSTALL SPACE (set by CMAKE_INSTALL_PREFIX)
  bin/
  etc/
  include/
  lib/
  share/
  .catkin
  env.bash
  setup.bash
  setup.sh
  ...
```

Рисунок 2.5 – Директории catkin_ws

Source space (src) содержит исходный код пакетов catkin. Из этого пространства можно извлечь исходный код для пакетов, которые необходимо построить. Каждая папка в src содержит некоторое количество пакетов catkin. Это пространство должно оставаться неизменным при настройке, построении или установке. Корень исходного пространства содержит символическую ссылку на файл CMakeLists.txt файла «toplevel» catkin. Этот файл вызывается CMake во время конфигурации проектов catkin в рабочей области.

В build space используется CMake для создания пакетов catkin в source space. CMake и catkin хранят информацию о КЭШе и другие промежуточные файлы. Build space не должно содержаться в рабочей области, и оно не должно находиться за пределами исходного пространства.

Development space — это место, где встроенные объекты помещаются перед установкой. То, как цели организованы в пространстве devel, должно совпадать с их компоновкой, когда они уже установлены. Это обеспечивает полезную среду тестирования и разработки. Расположение пространства devel контролируется определенной CMake-функцией Catkin с именем CATKIN_DEVEL_PREFIX и по умолчанию используется <build space> / develspace.

2) Multiple-sitl. Папка, содержащая эмуляцию прошивки автопилота и файлы для программного пакета Gazebo.

SITL (Software In The Loop) представляет собой сборку кода, который позволяет запускать модель без какого-либо дополнительного оборудования. Его можно создавать и запускать в операционных системах Windows или Linux, а также на Mac OSX (или на других платформах) или на виртуальной машине с установленной операционной системой Linux.

В этой папке находятся примеры физических сред, в которых можно тестировать модель коптера. На рисунке 2.6 представлен псевдокод загрузочного файла формата world.

```

<?xml version="1.0" ?>
<sdf version="1.4">
  <model name="angar">
    <static>true</static>
    <link name="link">
      <collision name="collision">
        <geometry>
          <heightmap>
            <uri>model://angar/materials/textures/heightmap.png</uri>
            <size>500 500 118</size>
            <pos>0 0 -15</pos>
          </heightmap>
        </geometry>
      </collision>
    </link>
    <visual name="visual">
      <geometry>
        <heightmap>
          <texture>
            <diffuse>file://media/materials/textures/dirt_diffusespecular.png</diffuse>
            <normal>file://media/materials/textures/flat_normal.png</normal>
            <size>5</size>
          </texture>
          <texture>
            <diffuse>file://media/materials/textures/grass_diffusespecular.png</diffuse>
            <normal>file://media/materials/textures/flat_normal.png</normal>
            <size>5</size>
          </texture>
          <texture>
            <diffuse>file://media/materials/textures/fungus_diffusespecular.png</diffuse>
            <normal>file://media/materials/textures/flat_normal.png</normal>
            <size>20</size>
          </texture>
          <blend>
            <min_height>15</min_height>
            <fade_dist>5</fade_dist>
          </blend>
          <blend>
            <min_height>30</min_height>
            <fade_dist>10</fade_dist>
          </blend>
          <uri>model://angar/materials/textures/heightmap.png</uri>
          <size>500 500 118</size>
          <pos>0 0 -15</pos>
        </heightmap>
      </geometry>
    </visual>
  </model>
</sdf>

```

Рисунок 2.6 – Псевдокод загрузочного файла формата world

3) Scripts. В этой папке лежит пример полета по точкам миссии example.py, а также пример обработки видеопотока - camera_node_example.py (пример подписывается на данные с одной из камер и делает преобразование бинаризации). example.py можно использовать как шаблон кода управления аппаратом. На рисунке 2.7 представлен псевдокод файла формата scripts

```

# coding=utf8
# Простая миссия. Вертикальный взлет, пролет вперед и посадка

height = 5.5

waypoints = [
    [2, 0, height],
    [2, 9, height],
    [30, 9, height],
    [50, 9, height],
    [55, 9, height]
]

```

Рисунок 2.7 – Псевдокод файла формата scripts

Каталог materials содержит все текстуры, изображения и сценарии OGRE для модели. Текстурные изображения должны быть помещены в подкаталог текстур и файлы сценариев OGRE в каталог сценариев. На рисунке 2.8 представлен внешний вид готовой имитационной твердотельной модели БПЛА.



Рисунок 2.8 – Внешний вид модели БПЛА

Компьютерная твердотельная модель беспилотного летательного аппарата представляет собой модель гексакоптера.

За основу твердотельной модели была взята модель typhoon_h480.

На рисунке 2.9 представлен псевдокод имитационной твердотельной модели БПЛА.

```

<sdf version="1.6">
  <model name="xfl">

    <pose>0 0 0 0 0 0 </pose>
    <link name="base_link">
      <pose frame="">0 0 0 0 -0 0</pose>
      <inertial>

        <pose frame="">0.001005 0 -0.0090035 0 -0 0</pose>
        <mass>2.02</mass>
        <inertia>
          <ixx>0.011</ixx>
          <ixy>0</ixy>
          <ixz>0</ixz>
          <iyy>0.015</iyy>
          <iyz>0</iyz>
          <izz>0.021</izz>
        </inertia>
      </inertial>
      <collision name="base_link_collision">
        <pose frame="">0 0 0.0 0 0 0</pose>
        <geometry>
          <box>
            <size>0.67 0.67 0.15</size>
          </box>
        </geometry>
        <surface>
          <contact>
            <ode>
              <min_depth>0.001</min_depth>
              <max_vel>0</max_vel>
            </ode>
          </contact>
          <friction>
            <ode />
          </friction>
        </surface>
      </collision>
      <visual name="base_link_visual">
        <pose frame="">0 0 0 0 -0 0</pose>
        <geometry>
          <mesh>
            <scale>0.001 0.001 0.001</scale>
            <uri>model://xfl/meshes/main_body_remeshed_v3.stl</uri>
          </mesh>
        </geometry>
      </visual>
    </link>
  </model>
</sdf>

```

Рисунок 2.9 – Псевдокод имитационной твердотельной модели БПЛА

На рисунке 2.10 и 2.11 представлена итоговая имитационная модель для исследования динамики движения мультикоптера.

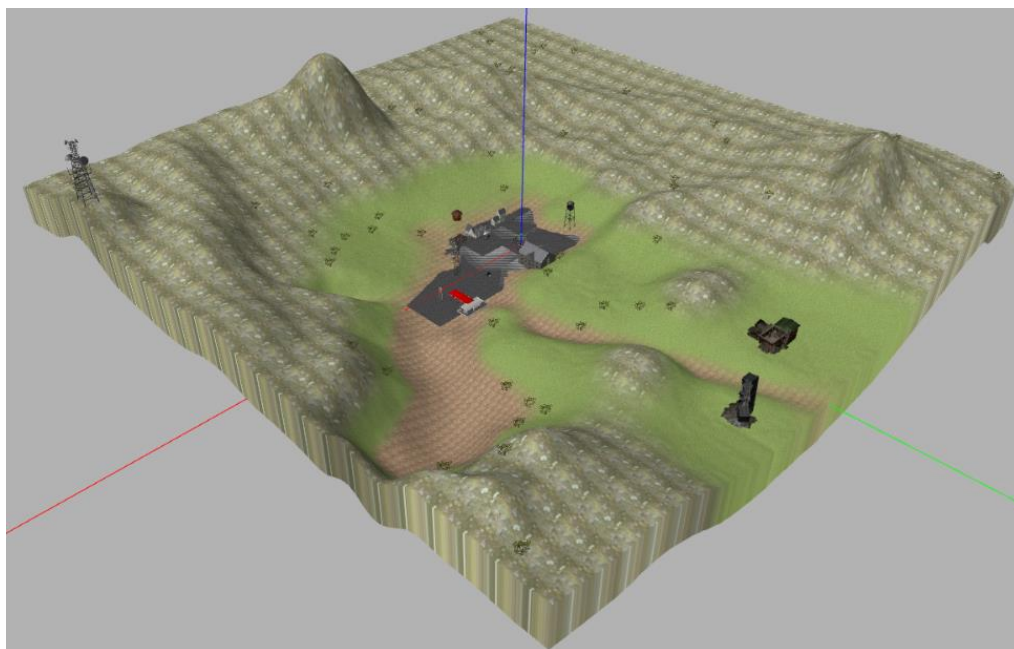


Рисунок 2.10– Модель окружающей среды в Gazebo

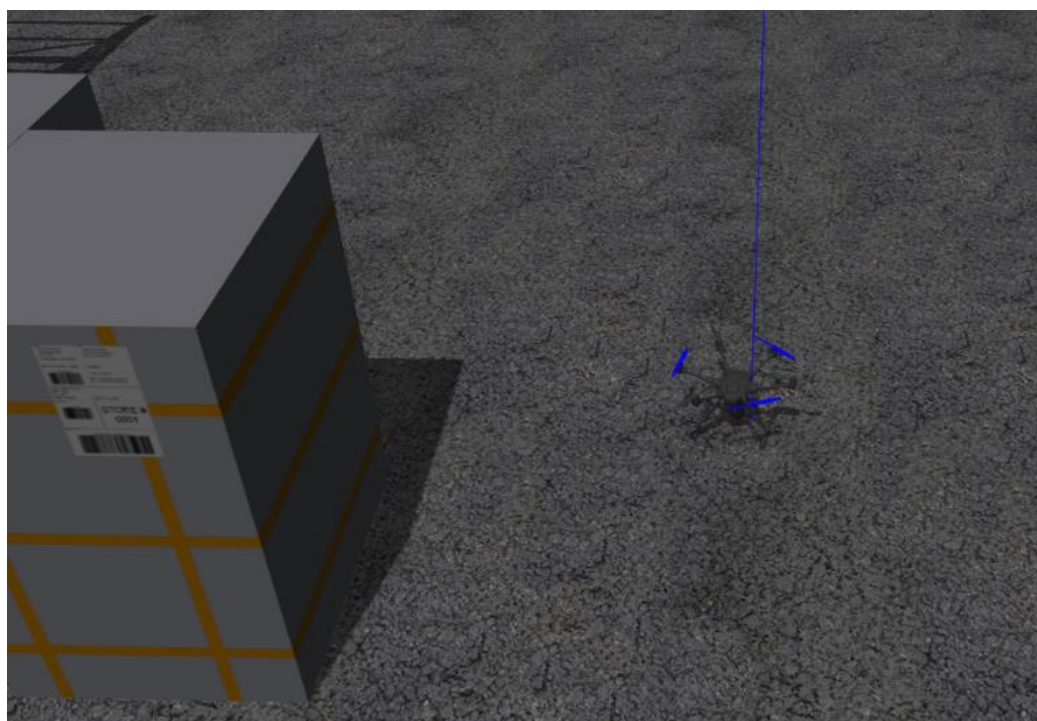


Рисунок 2.11 – Модель гексакоптера в имитируемой окружающей среде

Вывод: данные платформы обладают обширной документацией, бесплатным доступом, поддерживаются большим количеством устройств и имеют достаточный набор инструментов, ориентированных на робототехнику и разработку искусственного интеллекта.

2.2 Имитация работы лидара

Это метод съёмки, который измеряет расстояние до цели, освещая эту цель импульсным лазерным излучением и измеряя отраженные импульсы с помощью датчика. Различия во времени возврата лазера и длинах волн могут затем быть использованы для создания цифровых представлений цели.

Эта технология также используется для управления и навигации в имитационной модели БПЛА. Лидар использует инфракрасный, видимый или ближний инфракрасный свет для изображения объектов. Он может ориентироваться на широкий спектр материалов, включая неметаллические объекты, камни, дождь, химические соединения, аэрозоли, облака и даже отдельные молекулы. Узкий лазерный луч может отображать физические характеристики с очень высоким разрешением.

Velodyne Simulator позволяет нам моделировать Velodyne VLP-16 в Gazebo. Это программное обеспечение с открытым исходным кодом. Velodyne Simulator — это рабочее пространство для моделей, что означает, что он может быть построен с использованием команды `catkin build` в корневой папке пакета. Необходимо проделать определенную работу, чтобы работать с имитаторами лидаров. После постройки плагин и поиск его переменных в вашей среде, мы должны импортировать лидар к среде Gazebo.

Чтобы увидеть, работают ли датчики, мы можем повторить тему, например `rostopic echo / points_raw` или визуализировать данные датчика в RViz.

Правильная калибровка - важный аспект локализации и принятия решений чтобы извлечь реалистичную информацию и использовать ее. ROS реализовала библиотеку `tf`, которая была разработана для обеспечения стандартного способа сохранения и отслеживания координатных рамок и преобразования данных в пределах всей системы.

Запускаем трансформацию с лидара на базовую ссылку транспортного средства, используя узел `static_transform_publisher`. Первые три константы

аргументы args — это координаты x, y, z, а следующие три: roll, pitch and yaw в радианах. На рисунке 2.12 представлен узел static_transform_publisher.

```
1 <launch>
2   <node pkg="tf" type="static_transform_publisher" name="
   velodyne_to_base_link_tf" args="1.4 -0.5 1.75 0 0 0 1
   base_link velodyne 100" />
```

Рисунок 2.12 - Узел static_transform_publisher

Локализация - сопоставления отображаемых точек с текущими измеренными точками. Для локализации БПЛА необходимо запустить два узла.

Первый узел называется voxel_grid_filter. То, что делает этот узел, является понижающим показателем облака точек. Это означает, что оно уменьшает количество точек в наборе данных. Это делается по двум интуитивным причинам. Во-первых, поскольку лидар VLP-16 измеряет до 300 000 поинтов в секунду, будет много точек, которые будут измеряться очень близко один к другому.

Второй узел является фактическим узлом ndt_matching. Узлом ndt_matching является очень большой компонент - более 1600 строк кода. Он подписывается на шесть тем и публикует к тринадцати темам. Для наших целей наиболее важными темами являются ndt_matching . Узел публикует: / ndt_pose и / ndt_stat. Первая тема содержит сообщения, как следует из названия, о позе транспортного средства в окружающей среде. Тема / ndt_stat содержит сообщения по статистике расчетов - метку времени измерения, времени выполнения, итераций, необходимых для поиска результата, оценки, скорости транспортного средства и ускорение.

В качестве примера работы лидара запустим модель окружающей среды БПЛА без препятствий на пути и с препятствием. На рисунках 2.13 и 2.14 представлены имитации работы лидара без препятствий и с препятствиями соответственно.

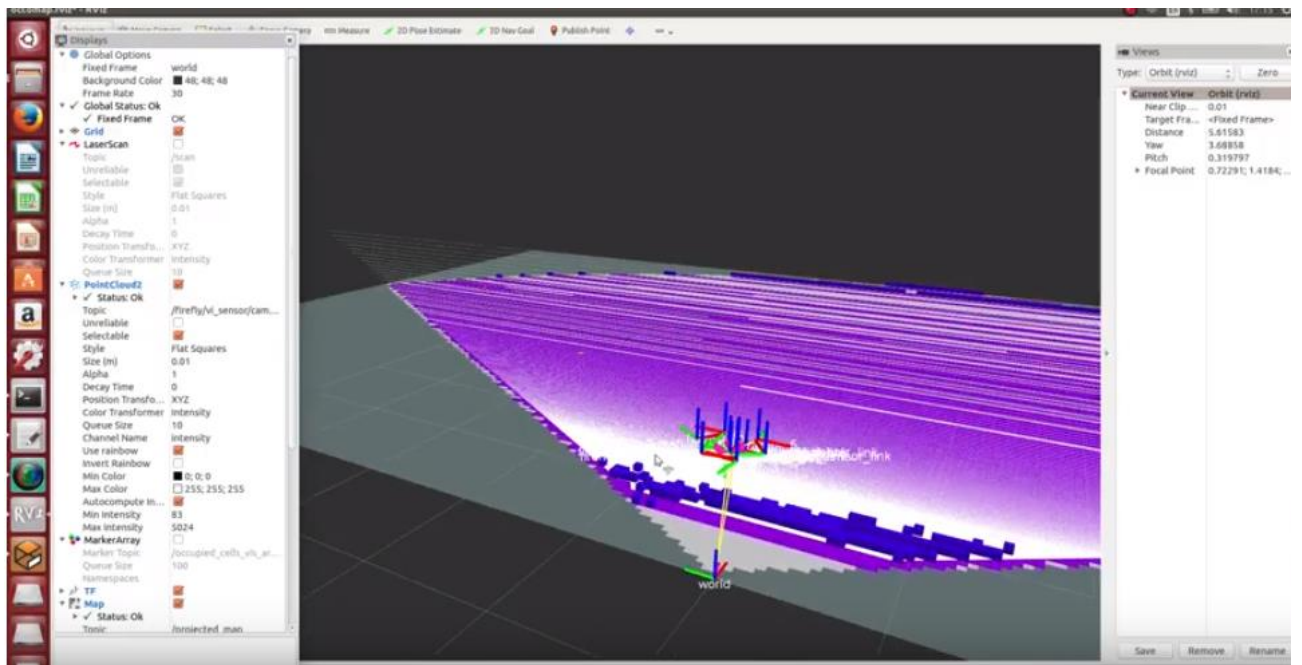


Рисунок 2.13 - Имитация работы лидара без препятствия на модели БПЛА

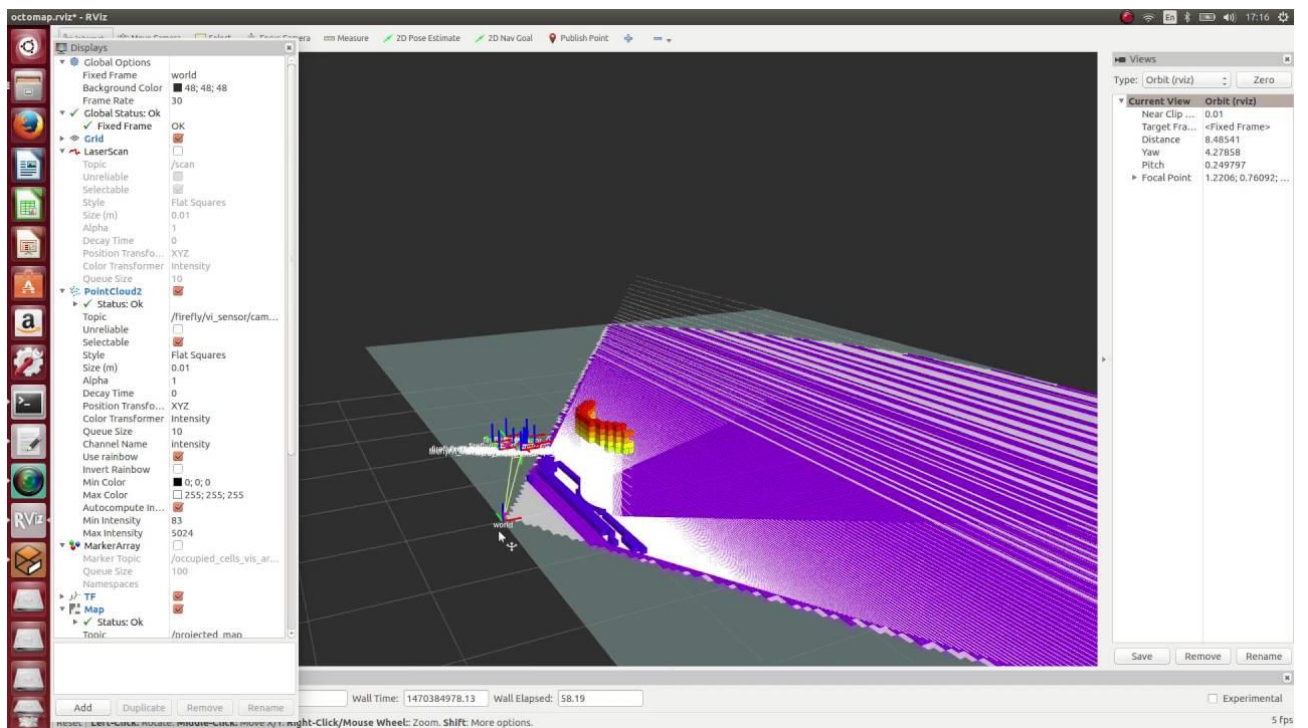


Рисунок 2.14 – Имитация работы лидара с препятствием на модели БПЛА

2.3 Имитация работы камеры

Gazebo поддерживает несколько типов плагинов, и все они могут быть подключены к ROS:

- 1) ModelPlugins, чтобы обеспечить доступ к физике: Model API;
- 2) SensorPlugins, чтобы обеспечить доступ к датчикам: API-интерфейс датчиков;
- 3) VisualPlugins, чтобы обеспечить доступ к визуализации API.

ModelPlugin вставляется в SDF внутри элемента `<robot>`. Он находится в тэге таблеткой `<gazebo>`, чтобы указать информацию, передаваемую в Gazebo. На рисунке 2.15 представлен пример псевдокода плагина Model.

```
<robot>
... robot description ...
<gazebo>
  <plugin name="differential_drive_controller" filename="libdiffdrive_plugin.so">
    ... plugin parameters ...
  </plugin>
</gazebo>
... robot description ...
</robot>
```

Рисунок 2.15 – Пример псевдокода плагина Model

Датчики в Gazebo предназначены для привязки к ссылкам, поэтому элемент `<gazebo>`, описывающий этот датчик, должен иметь ссылку на эту ссылку. На рисунке 2.16 представлен пример псевдокода плагина Sensor.

```
<robot>
... robot description ...
<link name="sensor_link">
  ... link description ...
</link>

<gazebo reference="sensor_link">
  <sensor type="camera" name="camera1">
    ... sensor parameters ...
    <plugin name="camera_controller" filename="libgazebo_ros_camera.so">
      ... plugin parameters ..
    </plugin>
  </sensor>
</gazebo>
</robot>
```

Рисунок 2.16 - Пример псевдокода плагина Sensor

После загрузки модели робота в Gazebo код `camera_controller` получит ссылку на датчик, обеспечивающий доступ к его API. Кроме того, он будет давать ссылку на сам SDF-элемент, чтобы прочитать переданные ему параметры плагина.

В качестве примера добавим к имитационной модели БПЛА камеру Microsoft Kinect. На рисунке 2.17 представлен псевдокод для камеры Kinect.

```
<gazebo>
  <plugin name="${link_name}_controller" filename="libgazebo_ros_openni_kinect.so">
    <baseline>0.2</baseline>
    <alwaysOn>true</alwaysOn>
    <updateRate>1.0</updateRate>
    <cameraName>${camera_name}_ir</cameraName>
    <imageTopicName>/${camera_name}/depth/image_raw</imageTopicName>
    <cameraInfoTopicName>/${camera_name}/depth/camera_info</cameraInfoTopicName>
    <depthImageTopicName>/${camera_name}/depth/image_raw</depthImageTopicName>
    <depthImageInfoTopicName>/${camera_name}/depth/camera_info</depthImageInfoTopicName>
    <pointCloudTopicName>/${camera_name}/depth/points</pointCloudTopicName>
    <frameName>${frame_name}</frameName>
    <pointCloudCutoff>0.5</pointCloudCutoff>
    <distortionK1>0.00000001</distortionK1>
    <distortionK2>0.00000001</distortionK2>
    <distortionK3>0.00000001</distortionK3>
    <distortionT1>0.00000001</distortionT1>
    <distortionT2>0.00000001</distortionT2>
    <CxPrime>0</CxPrime>
    <Cx>0</Cx>
    <Cy>0</Cy>
    <focalLength>0</focalLength>
    <hackBaseline>0</hackBaseline>
  </plugin>
</gazebo>
```

Рисунок 2.17 – Псевдокод для камеры Kinect

Камера Kinect представляет собой линейку датчиков движения, распознаваемых Microsoft для игровых консолей Xbox 360 и Xbox One и компьютеров Microsoft Windows. Основанный на периферийном модуле надстройки в стиле вебкамеры, он позволяет пользователям управлять и взаимодействовать со своей консолью/компьютером через естественный пользовательский интерфейс с использованием жестов и голосовых команд без необходимости присутствия игрового контроллера.

После добавления модели и сенсора камеры Kinect к модели БПЛА, становится возможным видеть через Rviz то, что видит на данный момент сам БПЛА. На рисунке 2.18 представлена Имитация работы камеры Kinect на модели гексакоптера.

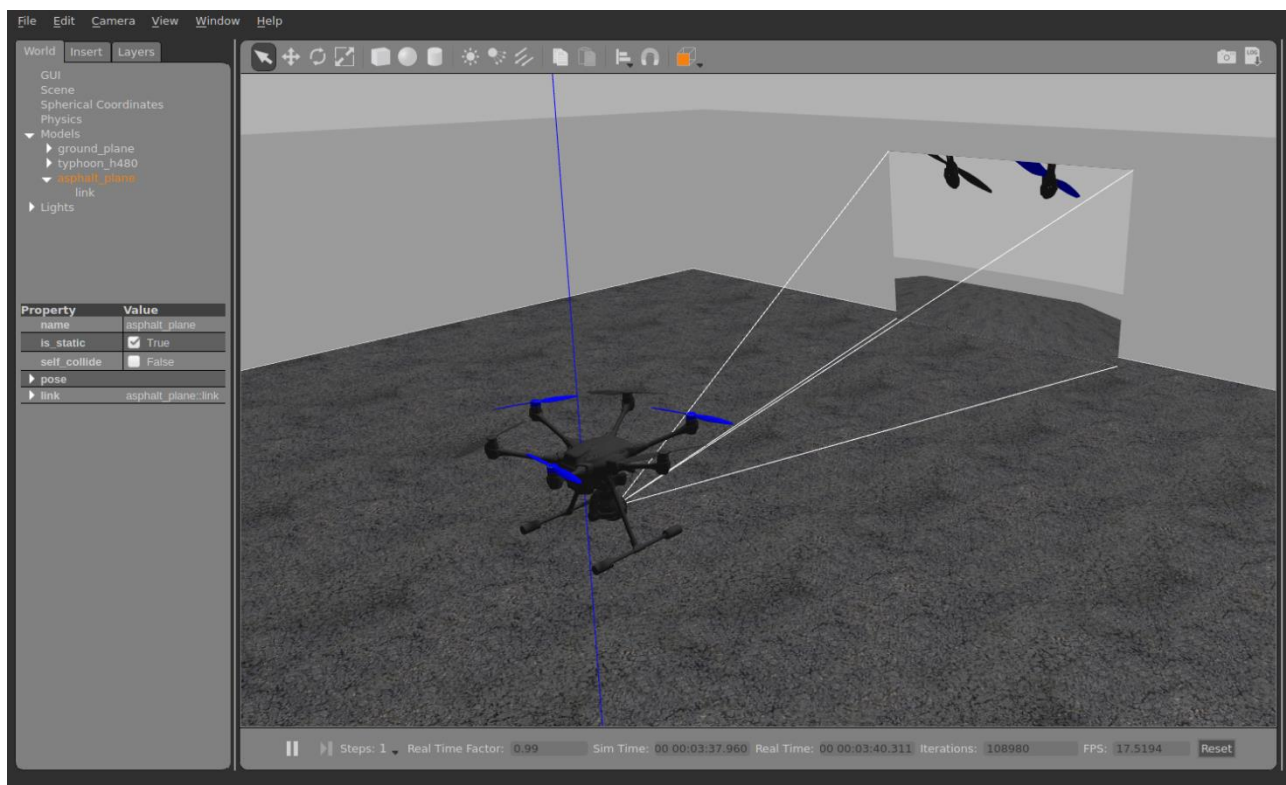


Рисунок 2.18 – Имитация работы камеры Kinect на модели гексакоптер

3 Основные этапы прочностного и аэродинамического анализа имитационной модели

3.1 Твердотельное моделирование

Разработка твердотельной модели БПЛА была выполнена в программном обеспечении SolidWorks. Для создания твердотельной модели были получены все геометрические размеры рабочей модели гексакоптера, затем разрабатывалась геометрия основных частей беспилотника.

Масса модели гексакоптера полностью соответствует массе оригинального устройства. Также стоит отметить, что итоговая масса объекта была получена как сумма масс отдельных деталей, которые также соответствуют массам оригинальных деталей [27].

Элементы сборки: винты, базы, моторы, лучи, крепления, сервоприводы, антенна, каркасы отдельных частей, держатели, диффузоры, шарниры, гиростабилизатор.

Каждый отдельный элемент сборки имеет свой собственный центр масс. Так как модель является симметричным телом, то центр масс гексакоптера полностью совпадает с геометрическим центром объекта.

Ниже представлены чертежи основных компонентов сборки гексакоптера и их 3D визуализация в пакете SolidWorks.

На рисунке 3.1 представлена 3D модель винта в SolidWorks.

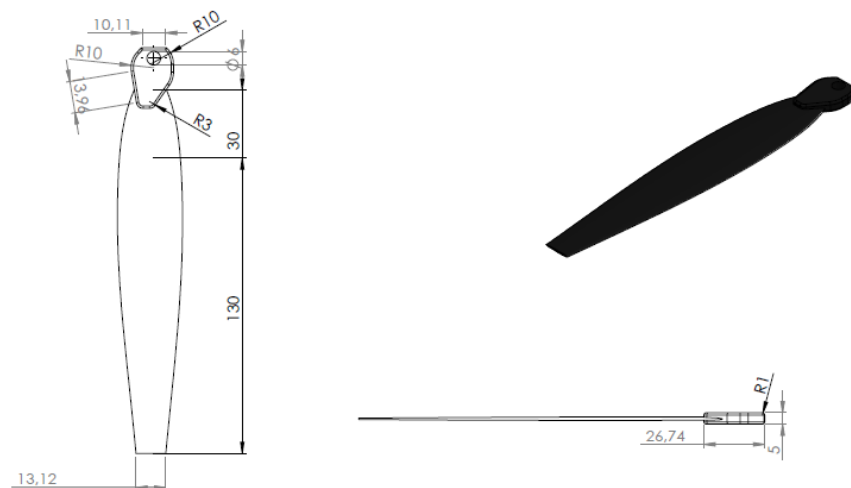


Рисунок 3.1 – 3D модель винта в SolidWorks

На рисунке 3.2 представлена 3D модель антенны в SolidWorks.

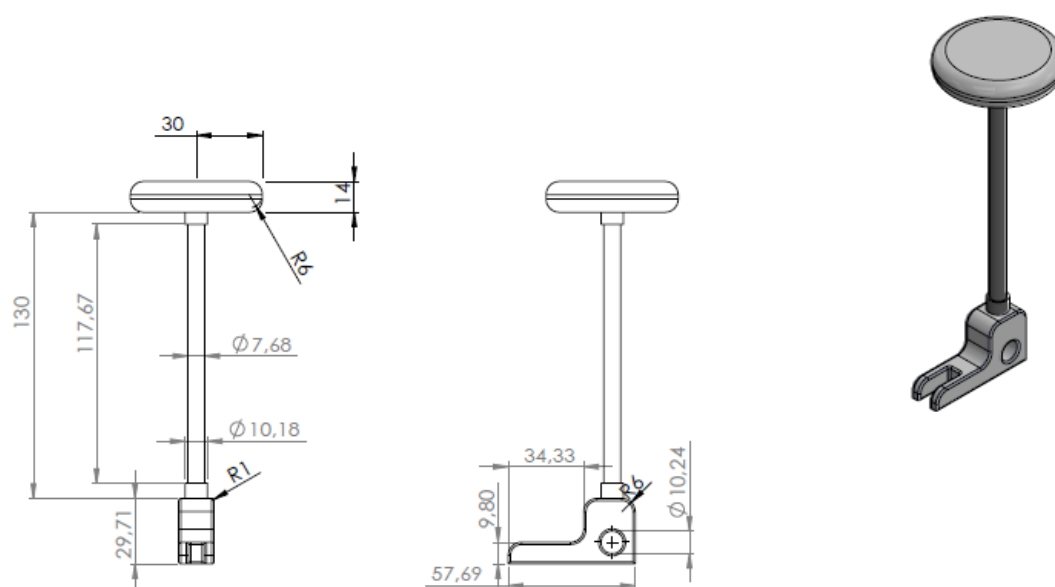


Рисунок 3.2 - 3D модель антенны в SolidWorks

На рисунке 3.3 представлена 3D модель каркаса мотора в SolidWorks.

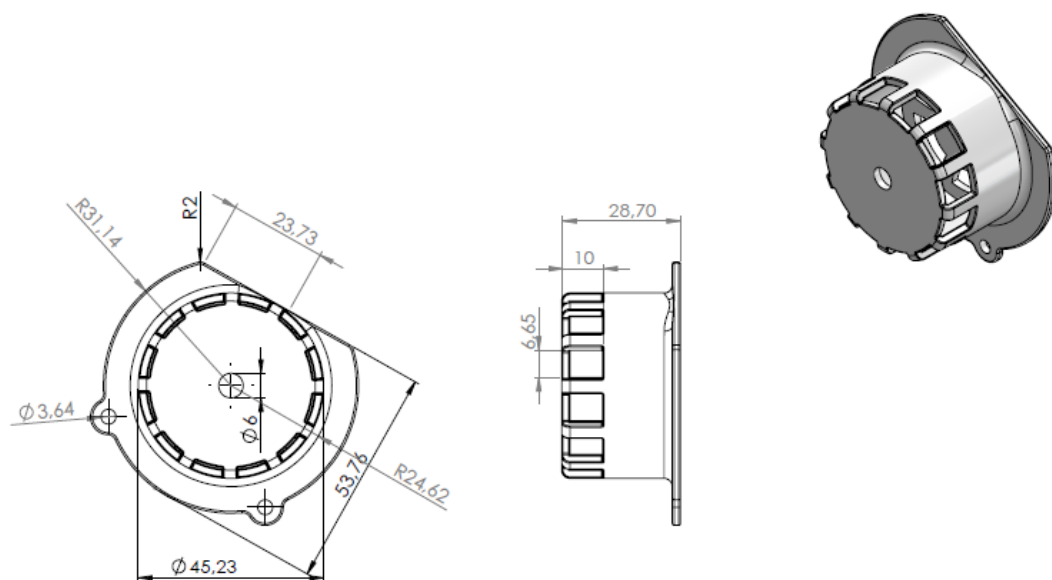


Рисунок 3.3 - 3D модель каркаса мотора в SolidWorks

На рисунке 3.4 представлена 3D модель мотора в SolidWorks.

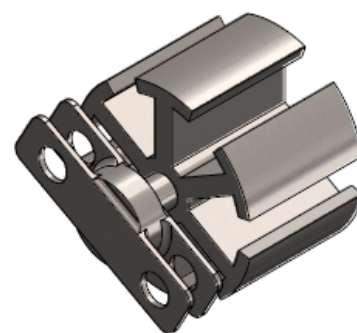
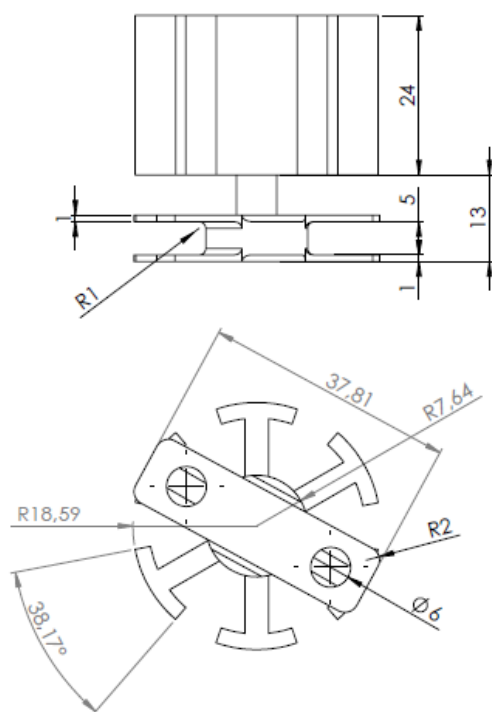


Рисунок 3.4 - 3D модель мотора в SolidWorks

На рисунке 3.5 представлена 3D модель держателя мотора в SolidWorks.

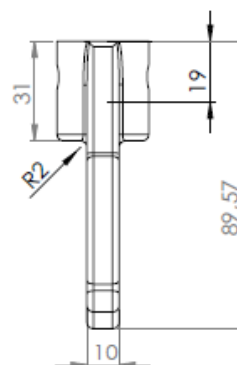
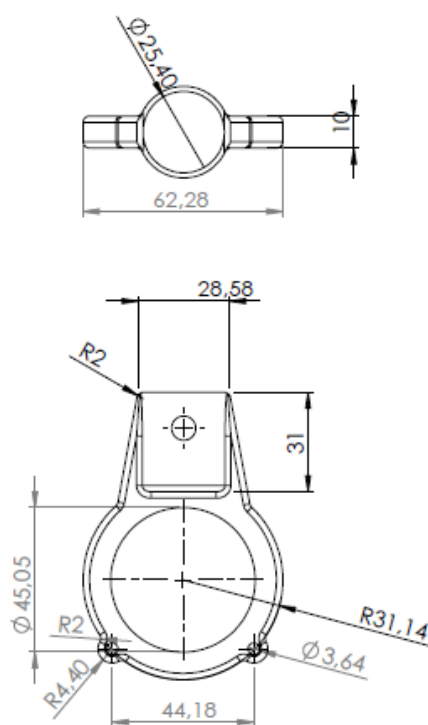


Рисунок 3.5 - 3D модель держателя мотора в SolidWorks

На рисунке 3.6 представлена 3D модель диффузора мотора в SolidWorks.

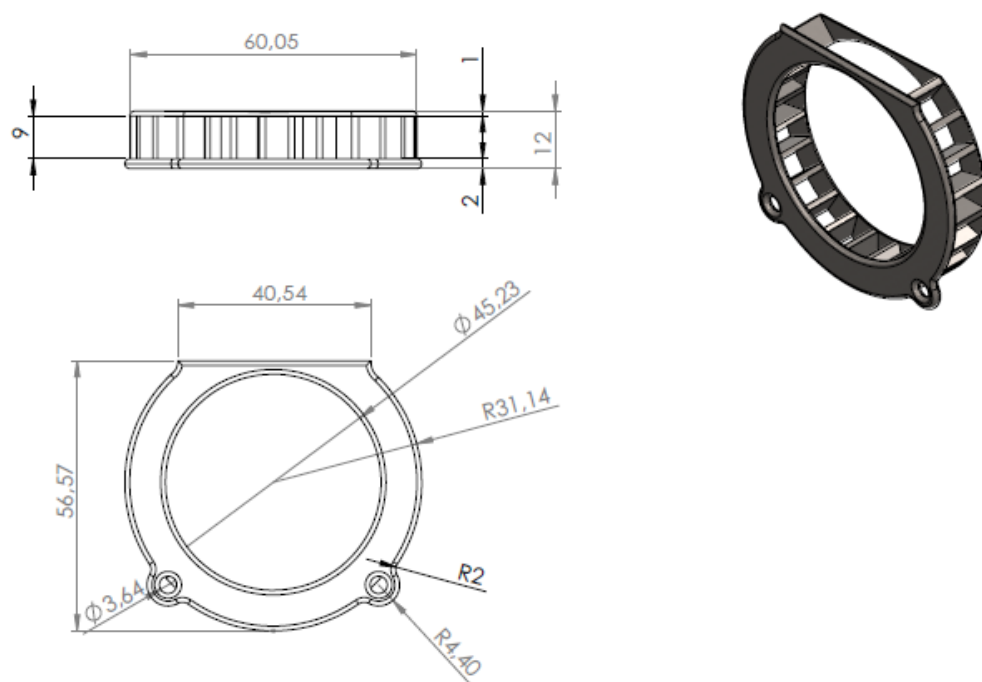


Рисунок 3.6 - 3D модель диффузора мотора в SolidWorks

На рисунке 3.7 представлена 3D модель лучей гексакоптера в SolidWorks.

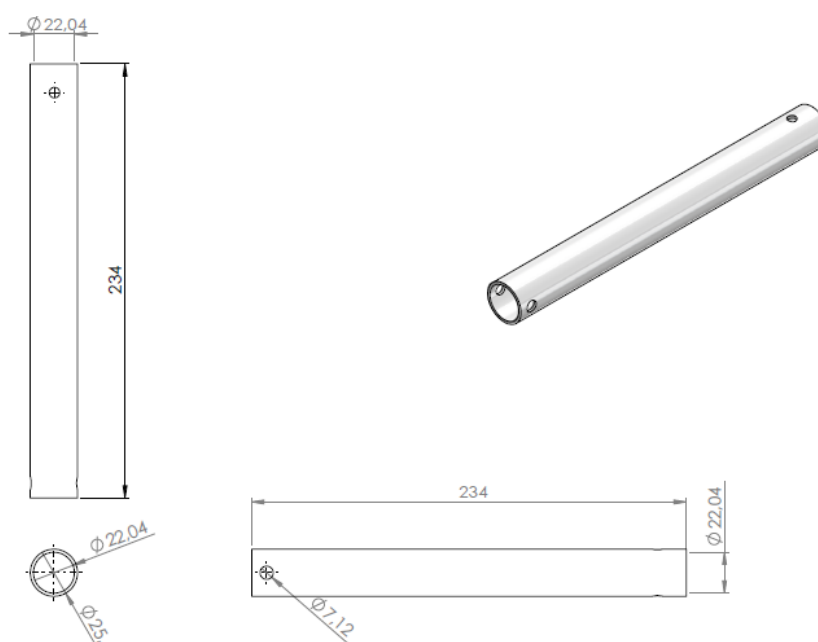


Рисунок 3.7 - 3D модель лучей гексакоптера в SolidWorks

На рисунке 3.8 представлена 3D модель держателя лучей гексакоптера в SolidWorks.

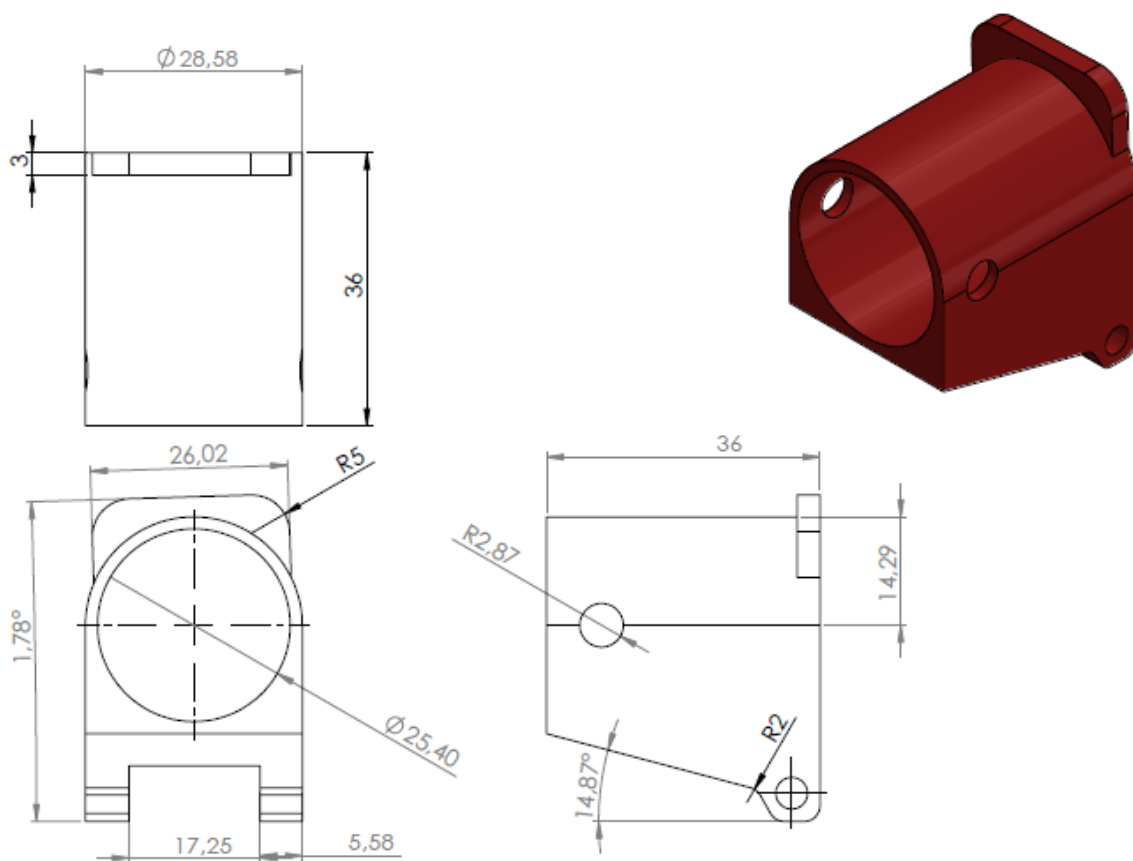


Рисунок 3.8 - 3D модель держателя лучей гексакоптера в SolidWorks

На рисунке 3.9 представлена 3D модель первой базы гексакоптера в SolidWorks.

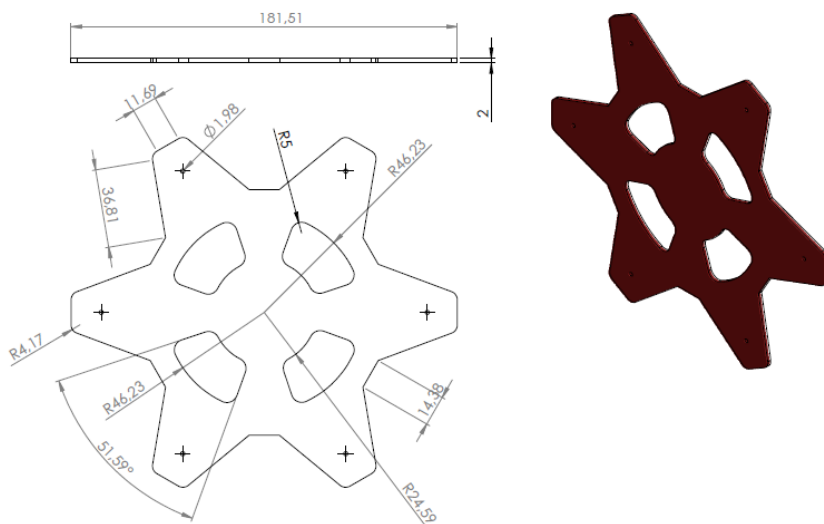


Рисунок 10.2.9 - 3D модель первой базы гексакоптера в SolidWorks

На рисунке 3.10 представлена 3D модель второй базы гексакоптера в SolidWorks.

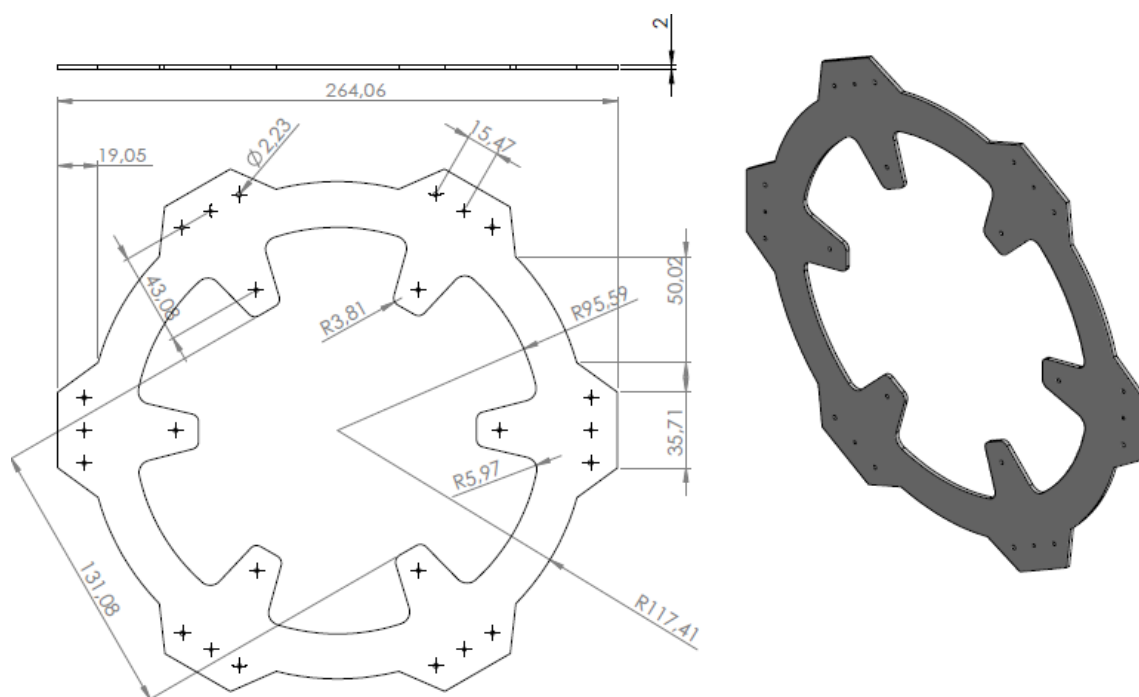


Рисунок 3.10 - 3D модель второй базы гексакоптера в SolidWorks

На рисунке 3.11 представлена 3D модель третьей базы гексакоптера в SolidWorks.

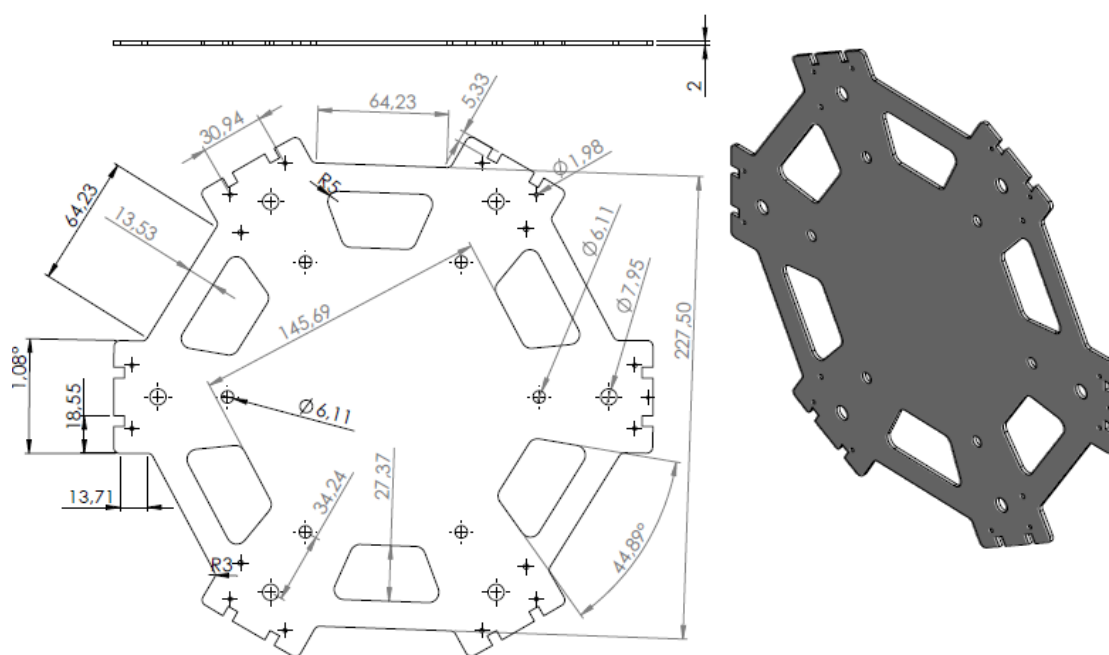


Рисунок 3.11 - 3D модель третьей базы гексакоптера в SolidWorks

На рисунке 3.12 представлена 3D модель четвертой базы гексакоптера в SolidWorks.

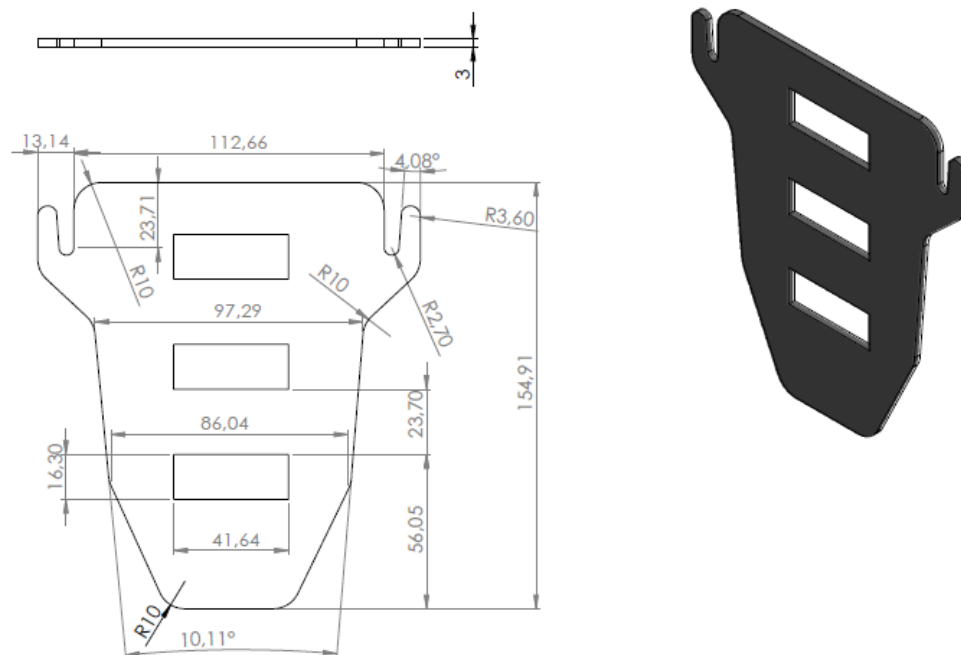


Рисунок 3.12 - 3D модель четвертой базы гексакоптера в SolidWorks

На рисунке 3.13 представлена 3D модель крепления баз гексакоптера в SolidWorks.

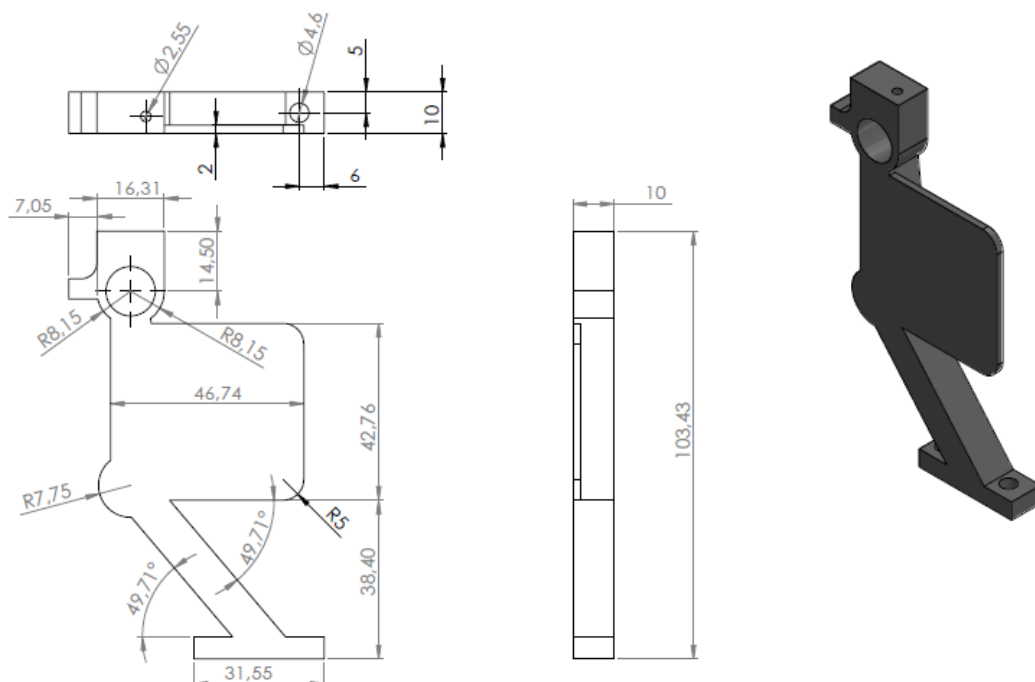


Рисунок 3.13 - 3D модель крепления баз гексакоптера в SolidWorks

На рисунке 3.14 представлена 3D модель сервопривод гексакоптера в SolidWorks.

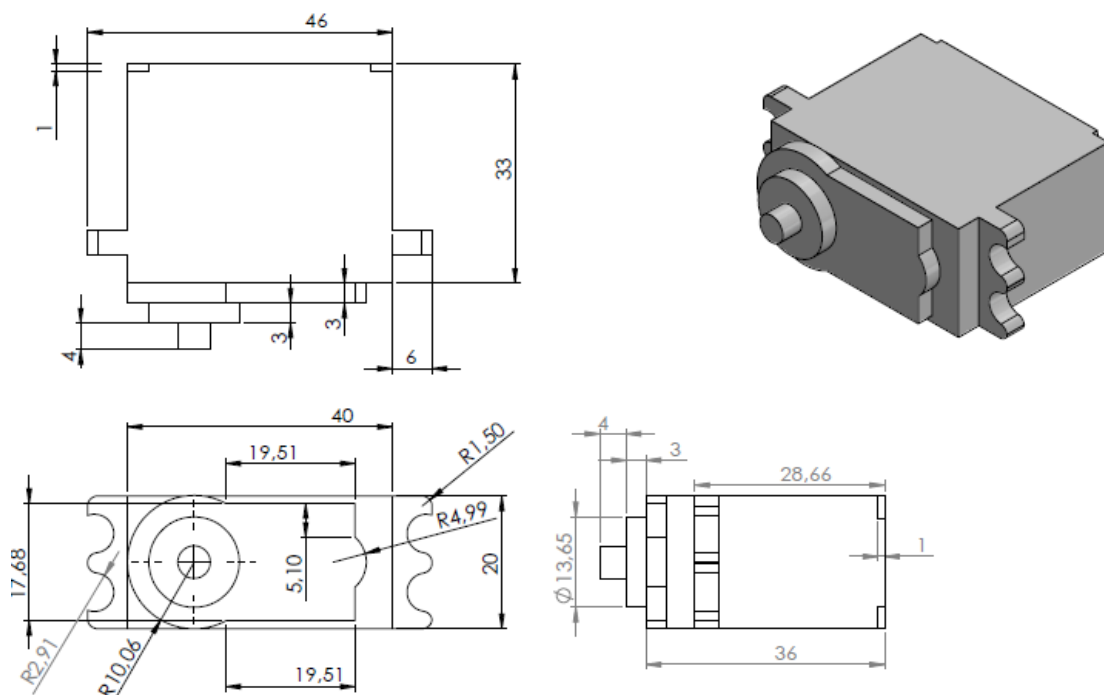


Рисунок 3.14 - 3D модель сервопривод гексакоптера в SolidWorks

На рисунке 3.15 представлена 3D модель креплений баз гексакоптера в SolidWorks.

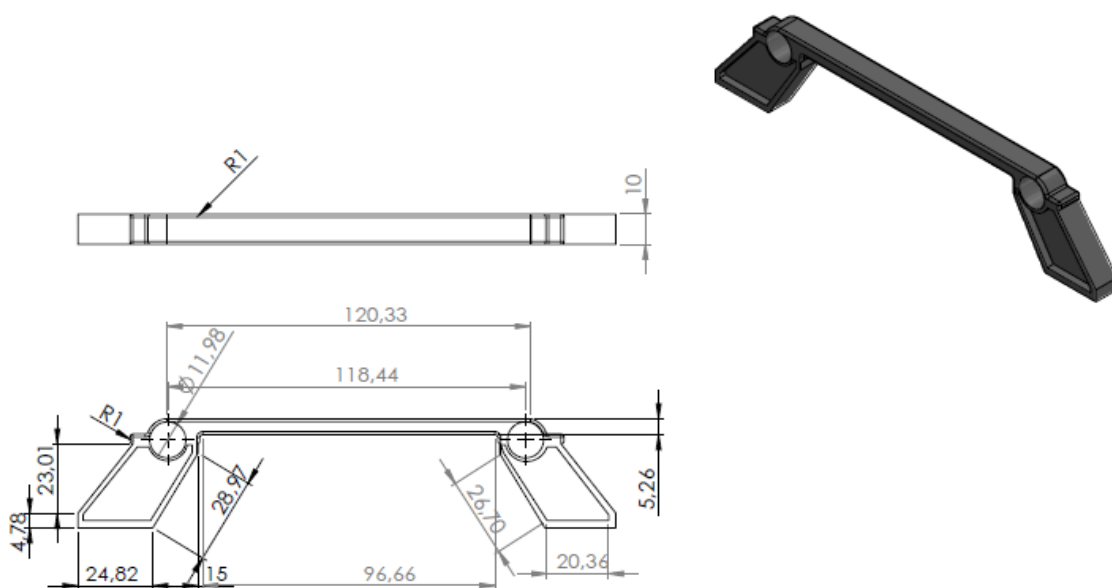


Рисунок 3.15 - 3D модель креплений баз гексакоптера в SolidWorks

На рисунке 3.16 представлена 3D модель шарнира гексакоптера в SolidWorks.

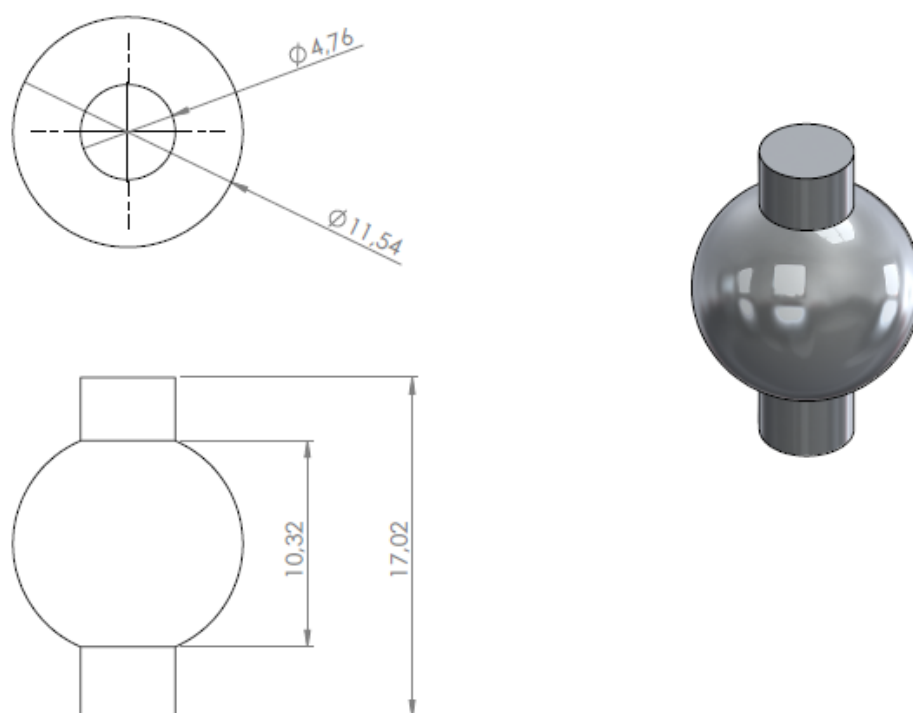


Рисунок 3.16 - 3D модель шарнира гексакоптера в SolidWorks

На рисунке 3.17 представлена 3D модель крепления баз гексакоптера в SolidWorks.

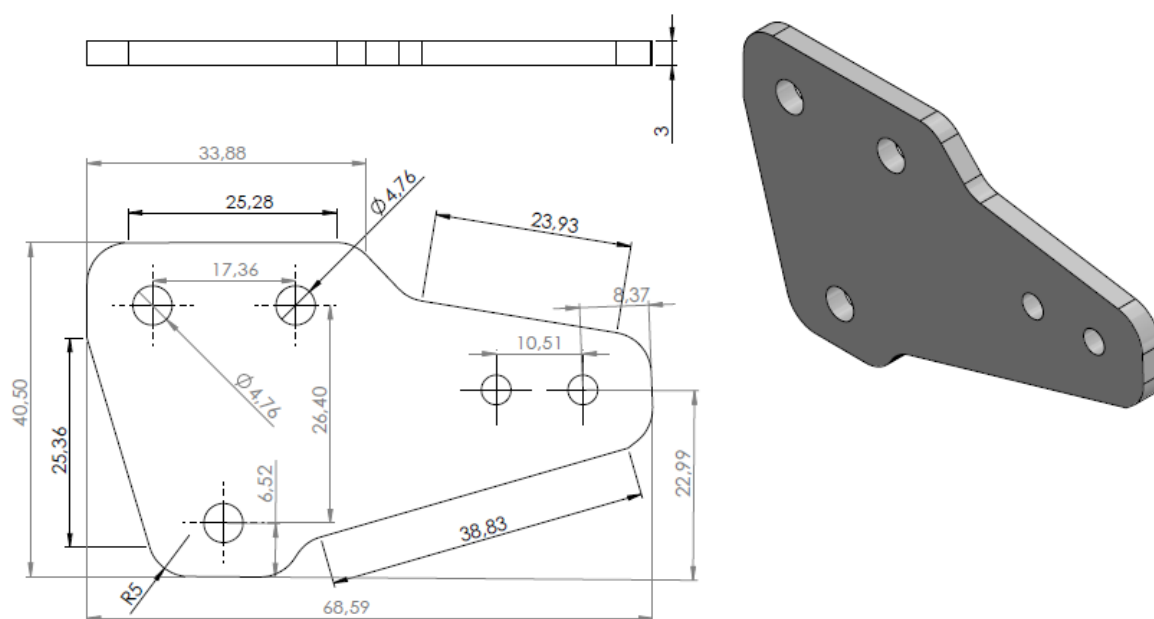


Рисунок 10.2.17 - 3D модель крепления баз гексакоптера в SolidWorks

На рисунке 3.18 представлена 3D модель гиростабилизатора Ronin MX для гексакоптера в SolidWorks.

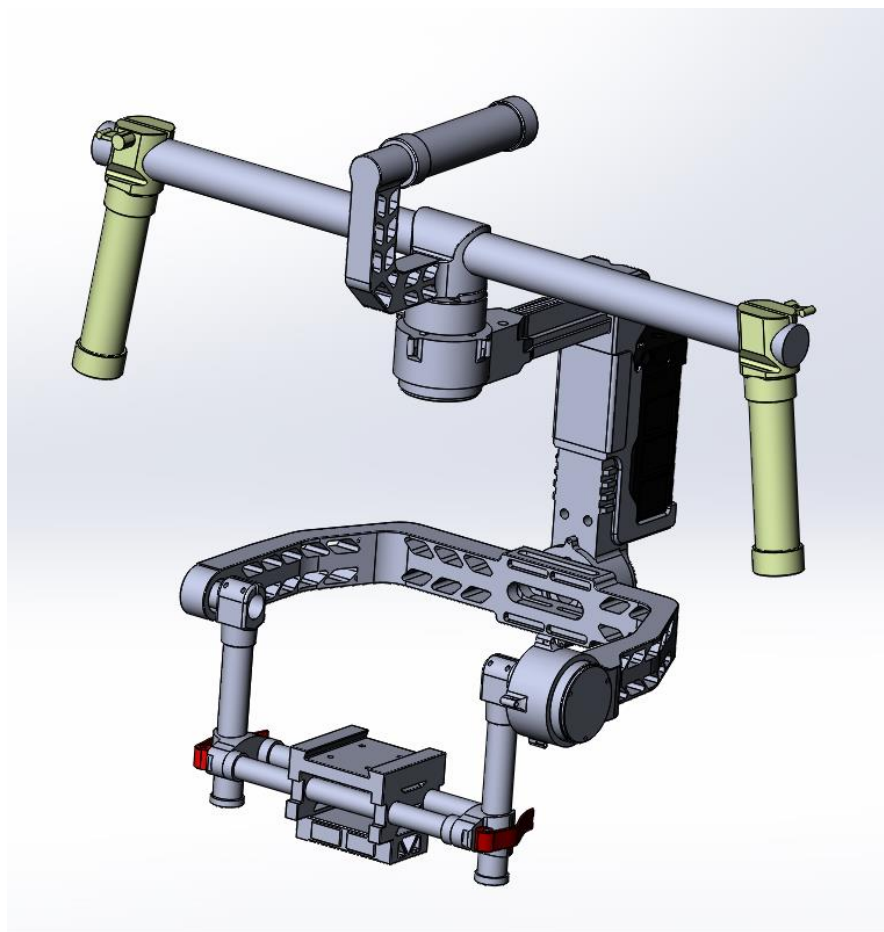


Рисунок 3.18 - 3D модель гиростабилизатора Ronin MX для гексакоптера в SolidWorks

На рисунке 3.19 представлено объединение частей 3D моделей гексакоптера в SolidWorks.



Рисунок 3.19 – Объединение частей 3D моделей гексакоптера в SolidWorks

На рисунке 3.20 представлена Готовая твердотельная 3D модель гексакоптера в SolidWorks.



Рисунок 3.20 – Готовая твердотельная 3D модель гексакоптера в SolidWorks

Вывод: программа SolidWorks считается одной из самых популярных систем автоматизированного проектирования, полностью удовлетворяющей потребности разнообразных организаций. Это мощный инструмент для комплексного проектирования изделий и компонентов любой сложности, в том числе и для промышленного дизайна. SolidWorks остается наиболее распространенным приложением для создания технических компонентов и твердотельного моделирования в целом.

3.2 Прочностной анализ имитационной модели

На первом этапе определения прочностных характеристик БПЛА заранее спроектированная трехмерная модель гексакоптера была импортирована в программу ANSYS Workbench 18.2.

На рисунке 3.21 представлена трехмерная модель гексакоптера.

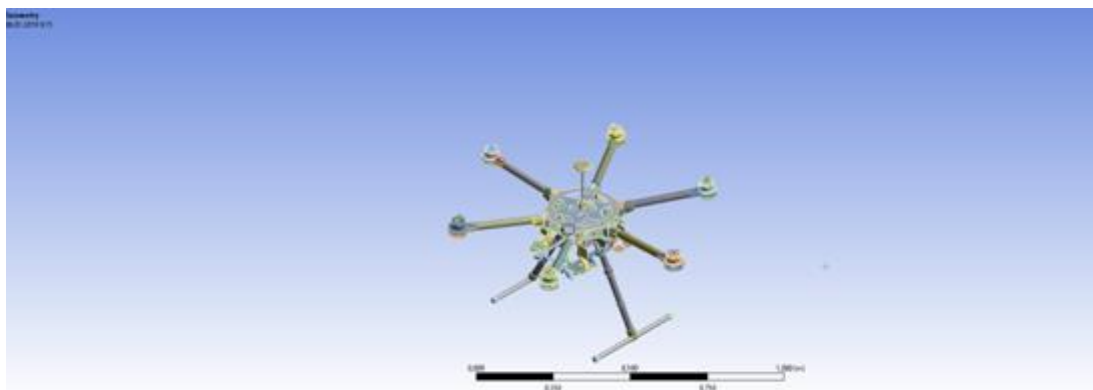


Рисунок 3.21 – Трехмерная модель гексакоптера

На следующем этапе было выполнено построение расчетной сетки. Задача построения расчетной сетки заключается в переводе узлов сетки физической области в вычислительную. Данное отображение, должно соответствовать следующим критериям [29]:

- отображение должно быть однозначным;
- сетка должна иметь сгущение в тех областях, где возможно появление больших градиентов искомых функций;

- линии сетки должны быть гладкими для обеспечения непрерывности производных;
 - сетки должны быть максимально близки к ортогональным (границы элементов сетки должны пересекать под углами близкими к 90°);
 - отношение сторон элемента сетки не должно быть слишком большим.
- На рисунке 3.22 представлена расчетная сетка гексакоптера.

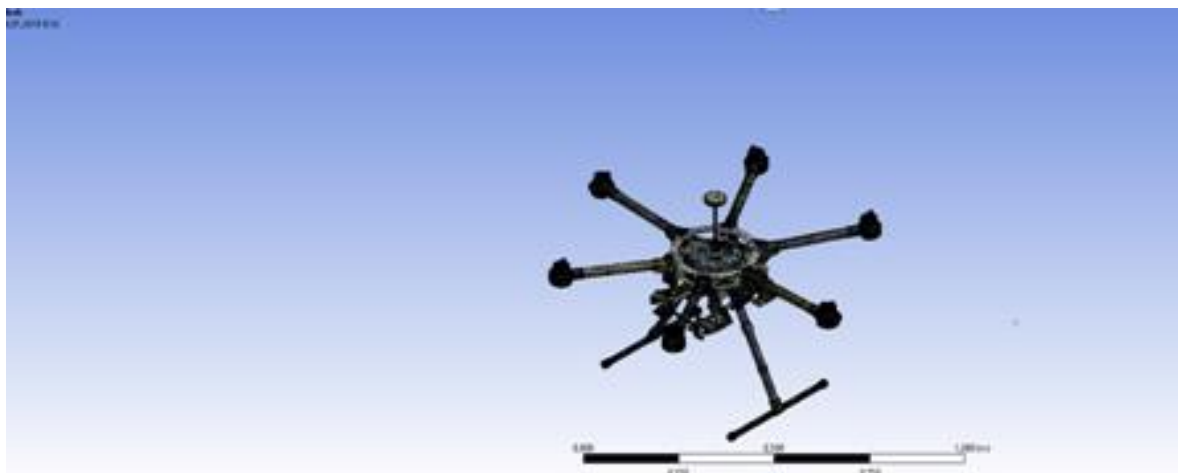


Рисунок 3.22 – Расчетная сетка гексакоптера

Расчет тяги винтов гексакоптера проводился в программе PropCalc. PropCalc вычисляет рабочие характеристики винтов с заданной геометрией, в частности, тягу и расход мощности в пределах доступного диапазона. PropCalc предоставляется с базой данных, которая содержит расчеты некоторых аэродинамических профилей, а также геометрические данные ряда некоторых винтов. На рисунке 3.23 представлены результаты расчета тяги винтов гексакоптера.

J	CT	CP	eta	Thr [N]	P [W]	v [m/s]	Re 0.7
0.05	0.0774	0.0244	0.16	23.008	374.9	2.6	168874
0.07	0.0758	0.0246	0.21	22.545	377.6	3.6	168914
0.09	0.0740	0.0247	0.27	22.017	379.5	4.6	168968
0.11	0.0721	0.0248	0.31	21.447	380.8	5.6	169034
0.13	0.0701	0.0249	0.36	20.856	381.4	6.6	169114
0.15	0.0680	0.0249	0.40	20.227	381.3	7.6	169206
0.17	0.0658	0.0248	0.44	19.575	380.7	8.6	169312
0.19	0.0635	0.0247	0.48	18.894	378.9	9.6	169431
0.21	0.0612	0.0245	0.51	18.187	376.3	10.6	169563
0.22	0.0587	0.0243	0.54	17.445	372.5	11.6	169707
0.24	0.0561	0.0240	0.57	16.688	367.8	12.6	169865
0.26	0.0535	0.0236	0.60	15.911	361.8	13.6	170035
0.28	0.0508	0.0231	0.62	15.121	354.8	14.6	170218
0.30	0.0481	0.0226	0.64	14.315	347.2	15.6	170414
0.32	0.0453	0.0220	0.66	13.481	338.2	16.6	170623
0.34	0.0425	0.0214	0.68	12.625	328.0	17.6	170845
0.36	0.0395	0.0206	0.69	11.749	316.6	18.6	171079
0.38	0.0365	0.0198	0.70	10.852	304.1	19.6	171326
0.40	0.0334	0.0189	0.70	9.932	290.2	20.6	171585
0.42	0.0303	0.0179	0.71	9.000	274.9	21.6	171857
0.44	0.0269	0.0169	0.70	8.011	258.7	22.6	172141
0.46	0.0235	0.0157	0.68	6.983	241.6	23.6	172438
0.48	0.0200	0.0146	0.65	5.948	223.3	24.6	172747
0.50	0.0164	0.0133	0.61	4.888	204.3	25.6	173088
0.52	0.0128	0.0120	0.55	3.805	184.2	26.6	173401
0.53	0.0091	0.0106	0.46	2.695	162.4	27.6	173747
0.55	0.0052	0.0093	0.31	1.544	142.1	28.6	174104
0.57	0.0013	0.0079	0.09	0.390	121.9	29.6	174474

Рисунок 3.23 – Результаты расчета тяги винтов гексакоптера

Исходя из проведенных расчетов, для того чтобы поднять гексакоптер с полезной нагрузкой массой 12 кг на высоту режима полета «зависание», необходимо иметь тягу для каждого винта равную 20 Н. В таблице 2 представлена задаваемая нагрузка на 3D модель гексакоптера [30].

Таблица 2 – Задаваемая нагрузка на 3D модель гексакоптера

Тип нагрузки	Значение
Стандартная сила тяжести	117,6 Н
Сила тяги винта 1	20 Н
Сила тяги винта 2	20 Н
Сила тяги винта 3	20 Н
Сила тяги винта 4	20 Н
Сила тяги винта 5	20 Н
Сила тяги винта 6	20 Н
Распределенная масса (полезная нагрузка)	8 кг

На рисунке 3.24 представлена задаваемая нагрузка на 3D модель гексакоптера.

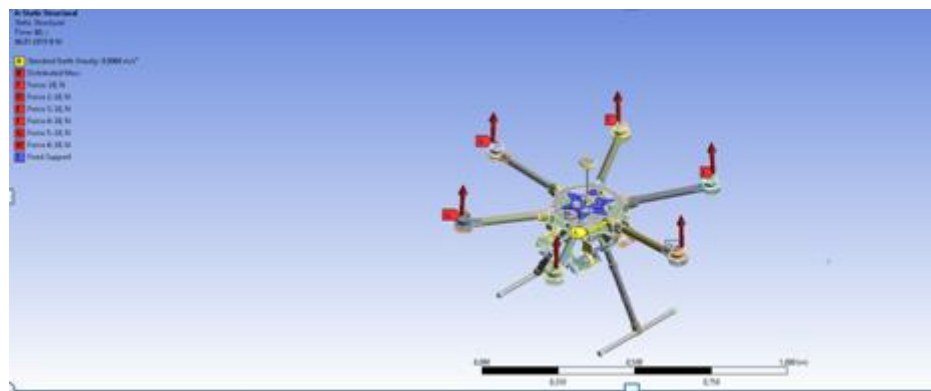


Рисунок 3.24 – Нагрузка на 3D модель гексакоптера

На рисунке 3.25 представлен результат определения напряжения в узлах конструкции гексакоптера.

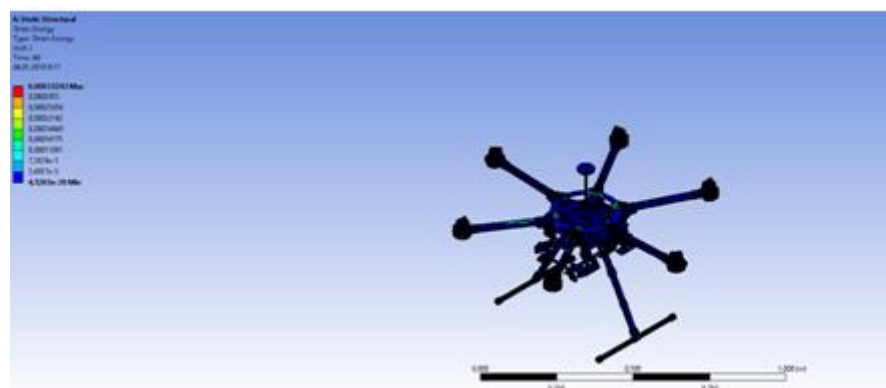


Рисунок 3.25 – Результат определения напряжения в узлах конструкции гексакоптера

На рисунке 3.26 представлен результат определения усталости конструкции гексакоптера.

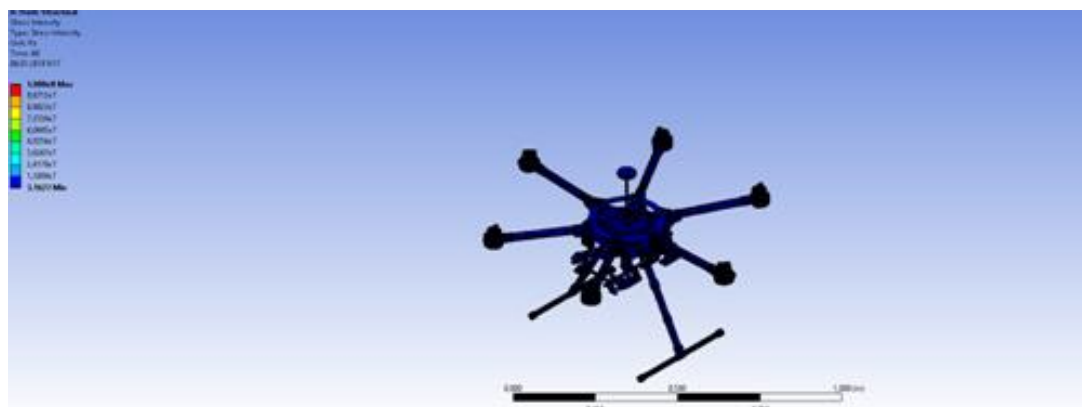


Рисунок 3.26 – Результат определения усталости конструкции
гексакоптера

На рисунке 3.27 представлен Результат определения деформации конструкции гексакоптера изометрия.

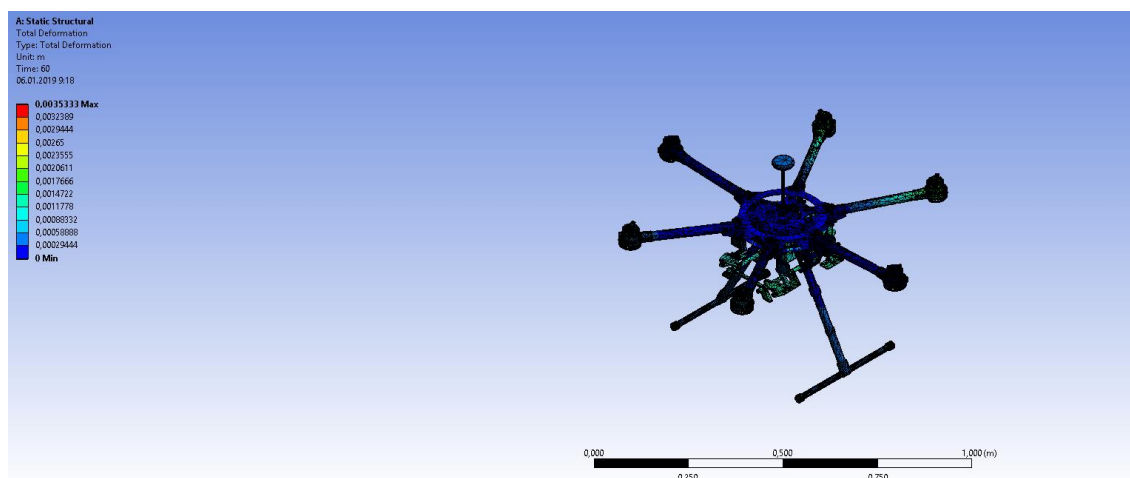


Рисунок 3.27 – Результат определения деформации конструкции
гексакоптера изометрия

На рисунке 3.28 представлен результат определения деформации конструкции гексакоптера, ось X.

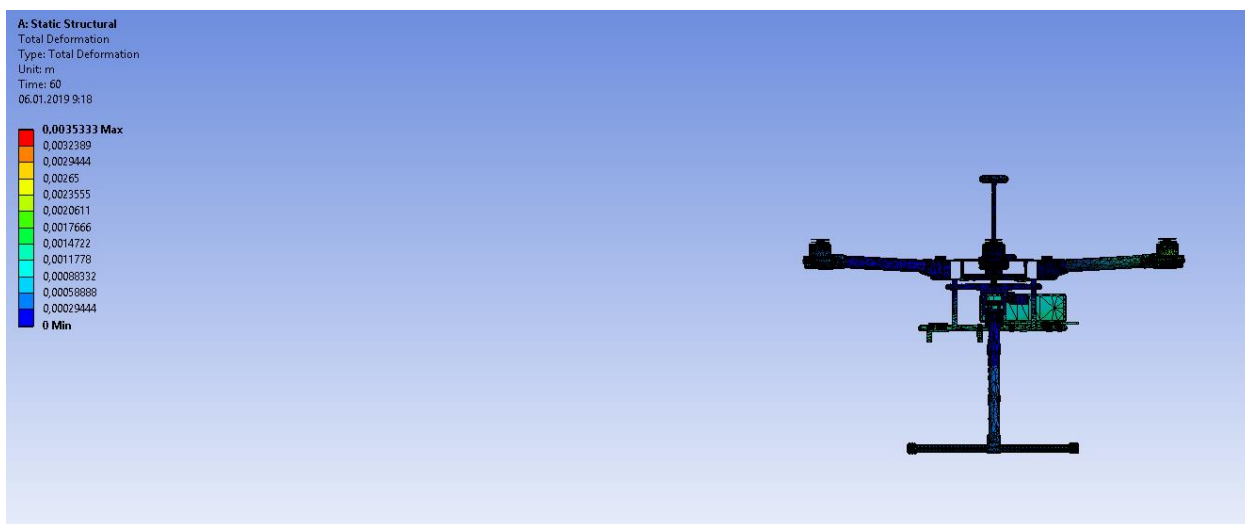


Рисунок 3.28 – Результат определения деформации конструкции гексакоптера, ось X

На рисунке 3.29 представлен Результат определения деформации конструкции гексакоптера, ось Y.

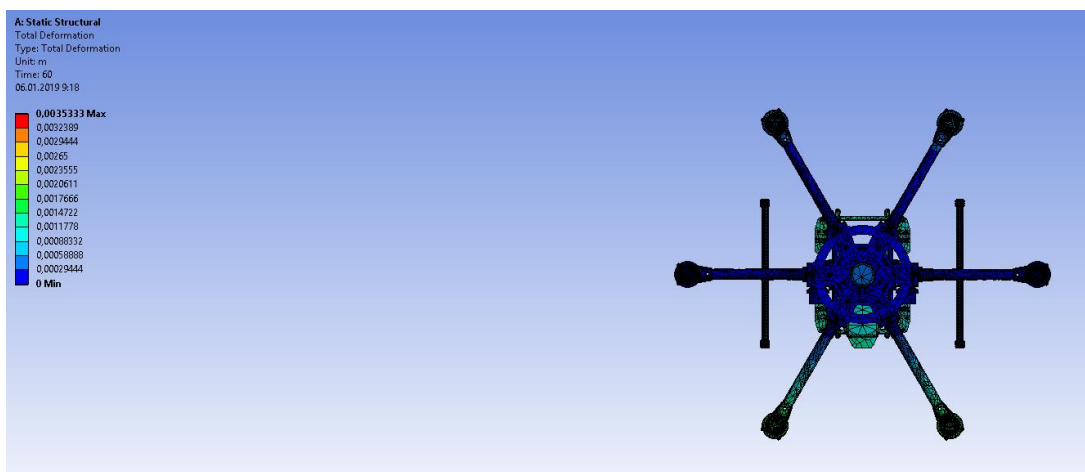


Рисунок 3.29 – Результат определения деформации конструкции гексакоптера, ось Y

На рисунке 3.30 представлен Результат определения деформации конструкции гексакоптера, ось Z.

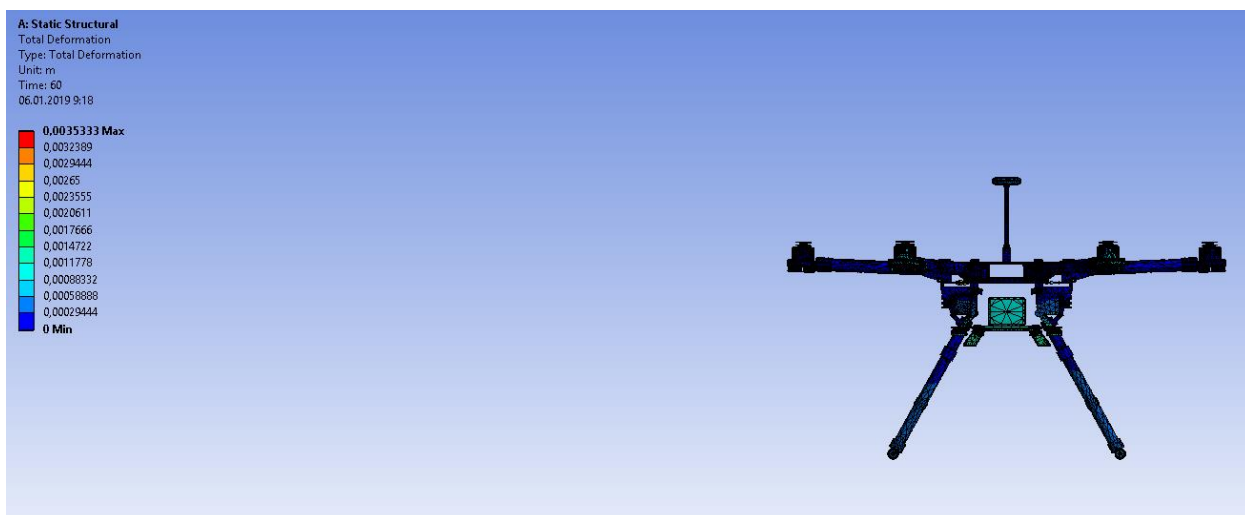


Рисунок 3.30 – Результат определения деформации конструкции гексакоптера, ось Z

На рисунке 3.31 представлен Результат определения деформации конструкции гексакоптера.

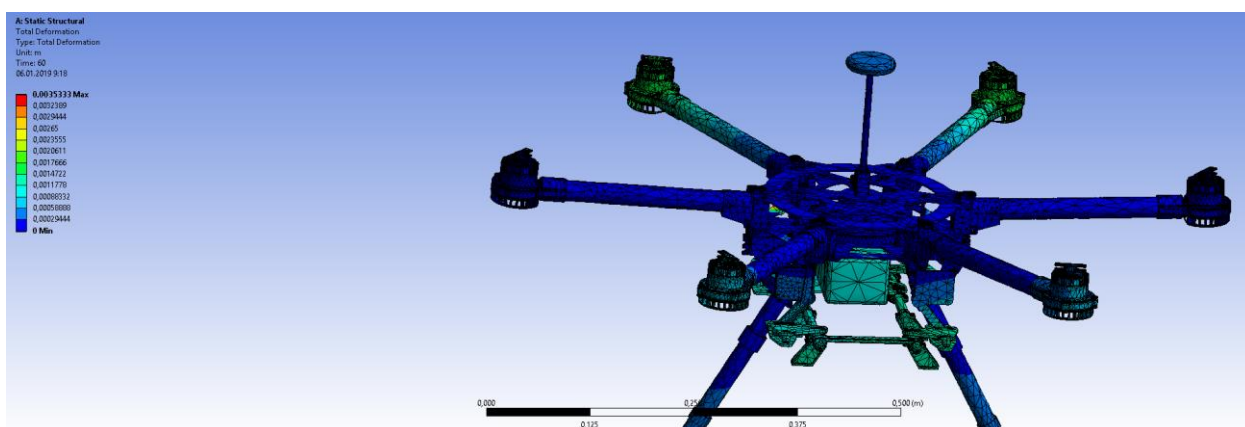


Рисунок 3.31 – Результат определения деформации конструкции гексакоптера

Вывод: как видно из решения разработанная конструкция гексакоптера будет обеспечивать необходимую прочность, и не разрушится, находясь, все время в пределах упругости.

3.3 Аэродинамический анализ имитационной модели

На сегодняшний день наиболее эффективным способом оценки аэродинамических свойств объекта является использование систем автоматизированного анализа (CAE). Особую роль среди существующих средств CAE играют комплексы метода конечных элементов. Данные комплексы позволяют проводить имитационное моделирование работы исследуемого физического явления, например, расчет аэродинамических свойств, на основе геометрии объекта, использования физики моделируемых процессов, а также свойств используемых материалов моделей. Среди комплексов МКЭ можно указать такие продукты, как ANSYS, ИСПА, COMSOL и др.

В магистерской диссертации для определения аэродинамических характеристик БПЛА используется программа ANSYS Workbench 18.2. ANSYS Workbench 18.2 позволяет в едином рабочем пространстве использовать несколько модулей одновременно для проведения связанного анализа. ANSYS предоставляет огромный выбор модулей для моделирования жидкостей и газов.

Выбор программы был основан на ее универсальности, так как ее можно использовать для решения большого количества задач: задач прочности, задач механики, задач электротехники, задач электромагнетизма, задач электроники, задач теплофизики, задач гидромеханики и задач биомедицины.

Исходными данными для программы являются математические модели изучаемых объектов (математические уравнения и граничные условия), геометрия исследуемых объектов, свойства материалов из которых состоит модель и наличие действующих на объект нагрузок.

Расчет аэродинамических характеристик БПЛА на первоначальном этапе предполагает построение модели исследуемого объекта в самой программе, либо в стороннем CAD пакете. Характеристики построенной

модели должны быть адекватны оригиналу, основываясь либо на физическом или информационном принципе.

Создание имитационной модели осуществляется в три этапа:

- разработка геометрической модели объекта;
- разработка физико-математической модели изучаемого процесса;
- разработка имитационной компьютерной модели.

Достоинством программного обеспечения ANSYS является то, что в программе обеспечивается возможность одновременной работы в режиме реального времени с твердотельным объектом и граничными условиями, благодаря большой вычислительной мощности программы.

Данный вид моделирования позволяет комбинировать простоту редактирования твердотельного объекта моделирования с возможностью построения дополнительных объектов.

В используемой программе существует набор стандартных интерфейсов. Каждый интерфейс имеет формат символьных файлов.

При этом каждый формат имеет свои определенные области применения. Стандартный формат DXF (Drawing eXchange Format) используется в основном для редактирования и построения чертежно-графической информации. Следующий формат IGES используется в основном для передачи геометрии поверхностных моделей.

В качестве импортируемой твердотельной модели БПЛА мультироторного типа использовался разработанная заранее геометрическая модель гексакоптера формата STEP, так как данный формат предоставляет в отличие от других форматов набор функций необходимых для решения поставленной задачи.

Предпочтение при выборе ПО для разработки геометрической модели было отдано среде SolidWorks.

Процесс выбора программного обеспечения для создания геометрии описан в главе 1.5. Выбор основывался на возможности построения адекватной модели с достаточным функционалом и возможностью

сохранения чертежа в формате STEP для дальнейшего импорта модели в ANSYS.

Исследуемым объектом является модель гексакоптера общие виды, которого представлены в разделе разработки модели в SolidWorks.

Для решения задач аэродинамики используется модуль ANSYS CFX. Данный модуль использует решатель, основанный на сетке конечных элементов (числовые значения в узлах сетки). Данный решатель предоставляет возможности физического моделирования определения аэродинамических характеристик для получения максимально точных результатов.

В настоящее время используют четыре основных направления для решения задач определения аэродинамических характеристик объекта:

- прямое численное моделирование. Решение нестационарных уравнений Навье - Стокса без каких-либо замыкающих соотношений;
- осредненные по Рейнольдсу уравнения Навье - Стокса замкнутые с помощью алгебраических или дифференциальных моделей турбулентности;
- метод крупных вихрей, который основан на решении нестационарных уравнений Навье - Стокса с моделированием влияния вихрей под сеточным масштабом;
- метод моделирования отсоединенных вихрей, который является комбинацией двух предыдущих подходов.

Модули решателя ANSYS CFX представлена на рисунке 3.32.

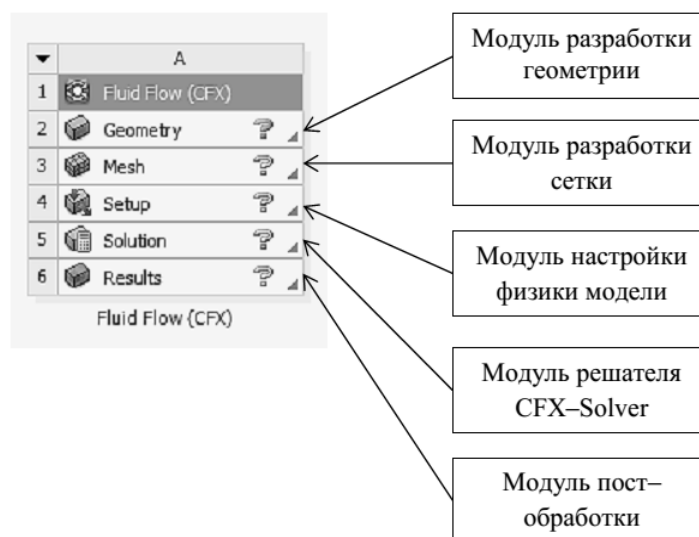


Рисунок 3.32 – Модули решателя ANSYS CFX

Структура расположения исследуемого проекта приведена на рисунке 3.33.

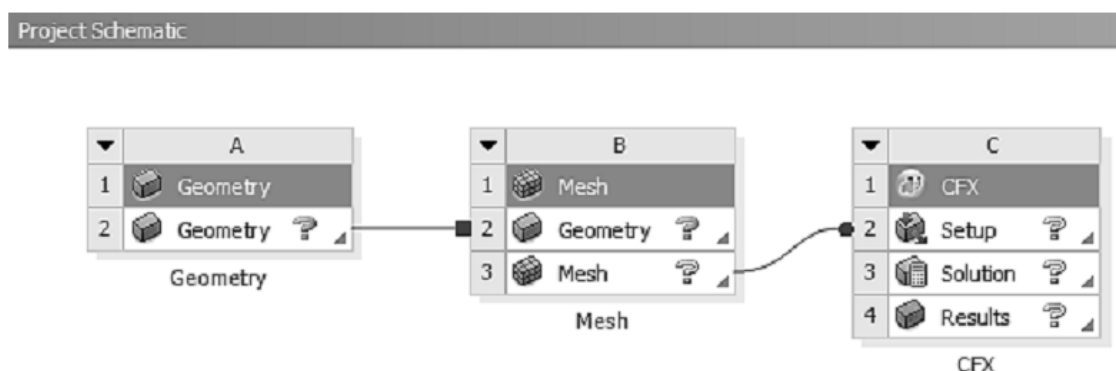


Рисунок 3.33 – Структура расположения исследуемого проекта:

A–импорт геометрии; *B* – создание расчетной сетки; *C* – расчет

С помощью модуля ANSYS Mesh на расчетную область накладывалась сетка конечных элементов. Для задач аэродинамики модуль ANSYS Mesh позволяет строить неструктурированные треугольные и четырехугольные поверхностные сетки, учитывающие кривизну и сближение поверхностей, при этом обеспечивается плавное изменение размеров элементов и их высокое качество. Построение поверхностных сеток методами неструктурированных треугольных и четырехугольных поверхностей

является необходимым для исследуемой модели гексакоптера из-за наличия кривизны и мелких деталей. Для построения сеток несущих винтов используется метод построения тетраэдрами.

Особенностью используемого расчётного метода является его широкая информативность. Метод CFX позволяет не только получать суммарные аэродинамические характеристики и распределение давления по контуру профиля, но и проводить визуальный анализ обтекания профиля и окружающего его пространства. Такой подход даёт возможность выявить участки с неблагоприятным срывным характером обтекания и сузить количество экспериментальных исследований при выборе оптимального набора профилей по модели гексакоптера. Кроме того, информация о действующих распределённых нагрузках по контуру профиля является важной для прочностных расчётов элементов конструкции гексакоптера на стадии проектирования.

Расчётная сетка профиля имеет блочную структурированную топологию. Результат построения поверхностной сетки представлен на рисунке 3.32 и 3.35.

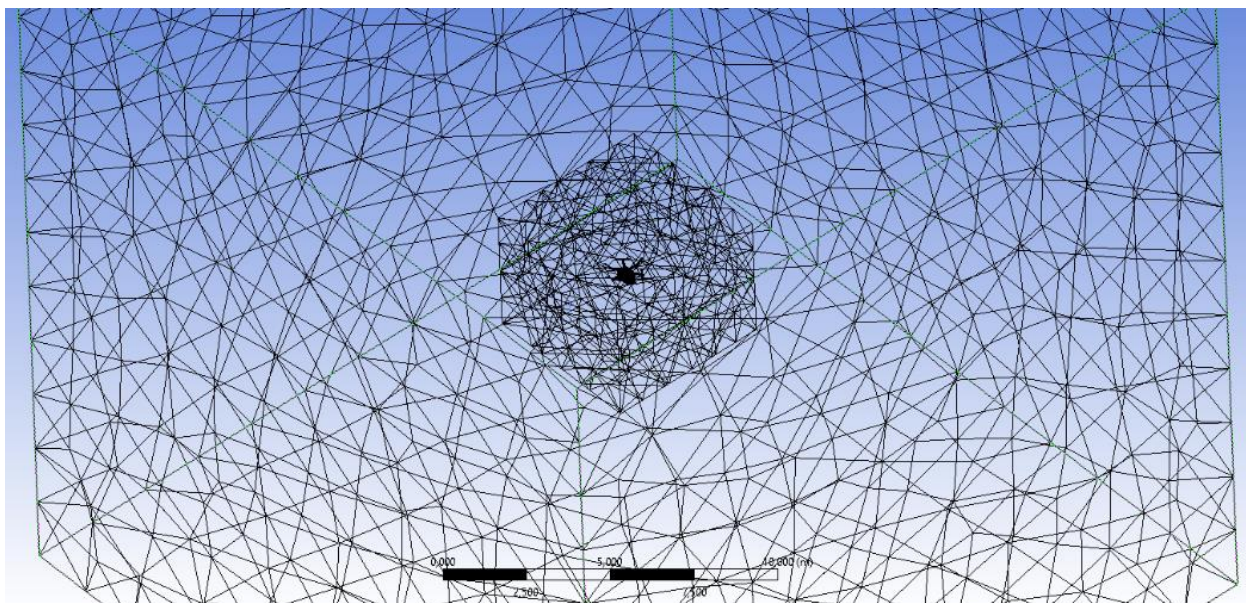


Рисунок 3.34 – Расчетная поверхностная сетка имитационной модели гексакоптера

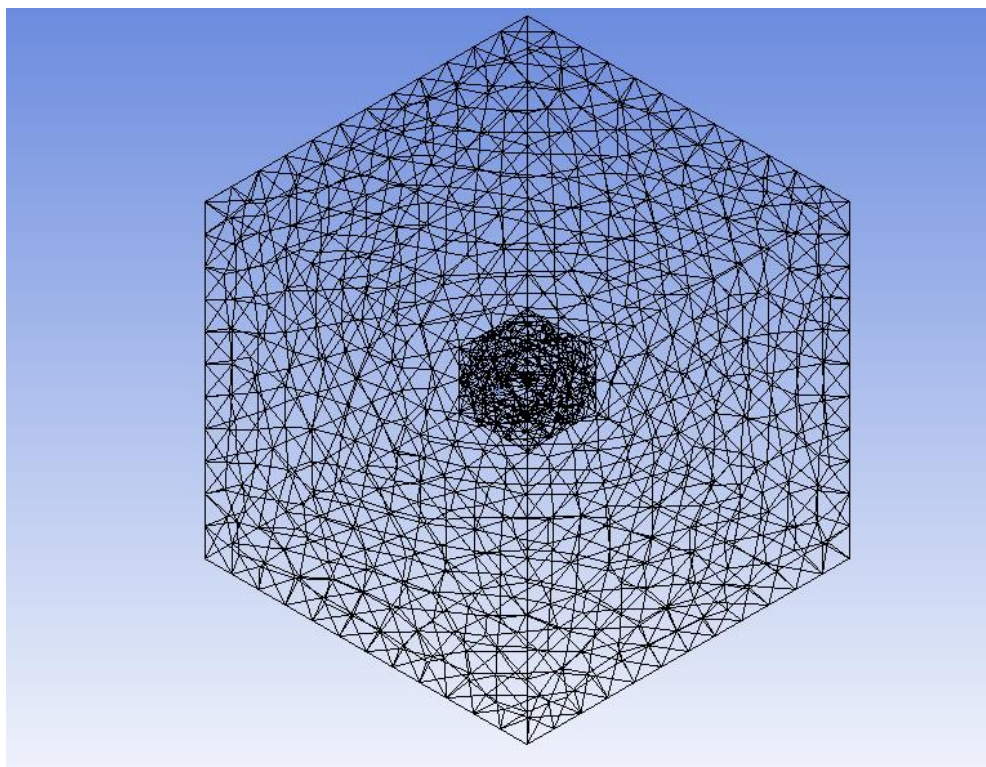


Рисунок 3.35 – Расчетная поверхностная сетка имитационной модели
гексакоптера, изометрия

После построения расчетной поверхностной сетки имитационной модели модель загружается в подмодуль Setup для задания условий определения аэродинамических свойств модели гексакоптера. На рисунке 3.36 представлена структура подмодуля Setup.

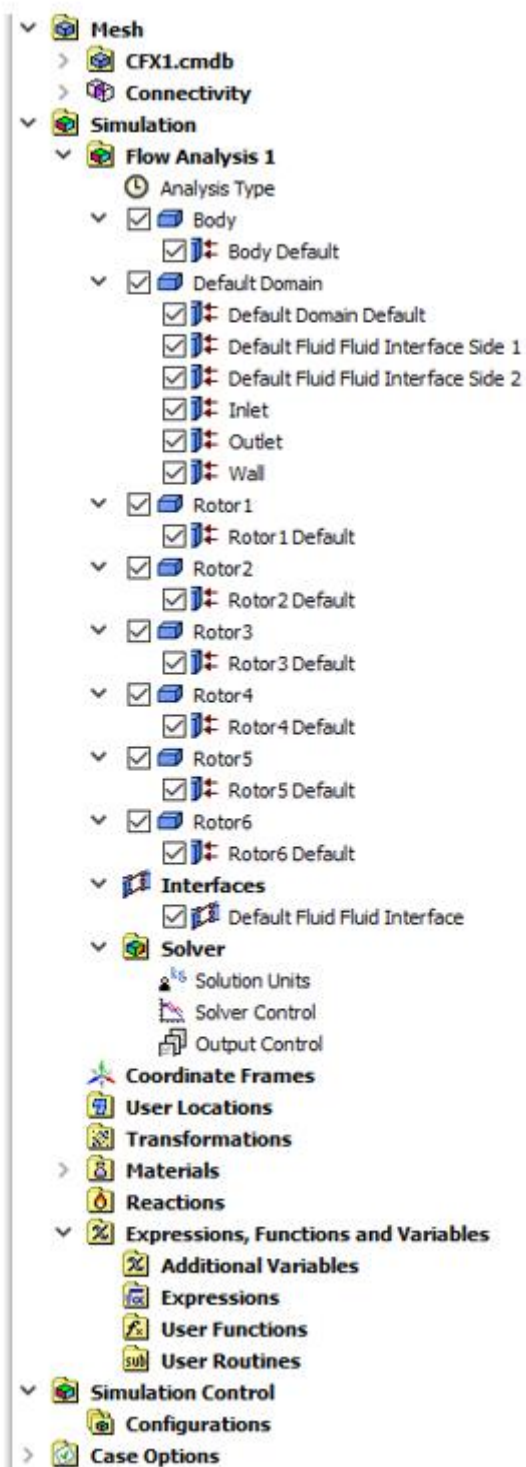


Рисунок 3.36 – Структура подмодуля Setup

Моделирования аэродинамических свойств беспилотного летательного аппарата производится при его движении в окружающем пространстве с диапазоном изменения скорости от 1 до 150 км/ч.

Для наглядности определения аэродинамических свойств модели гексакоптера были произведены расчеты при следующих скоростях: 0 км/ч, 30 км/ч, 60 км/ч, 100 км/ч, 150 км/ч. На рисунке 3.37 представлена модель задания условий определения аэродинамических свойств исследуемой имитационной модели гексакоптера.

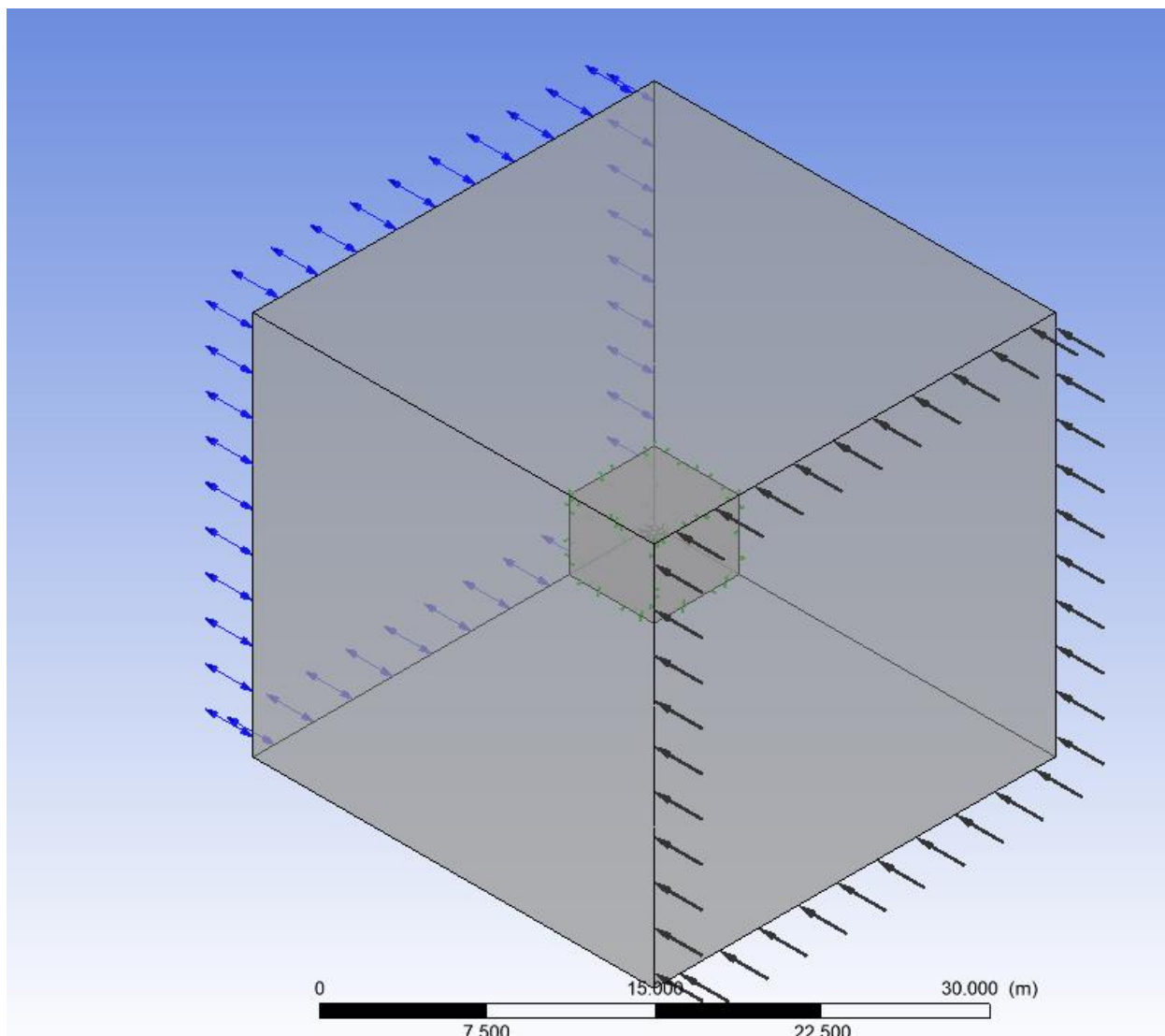


Рисунок 3.37 – Модель задания условий определения аэродинамических свойств исследуемой имитационной модели гексакоптера

На рисунке 3.38 представлена Модель задание условий для входной стенки окружающего пространства исследуемой имитационной модели гексакоптера.

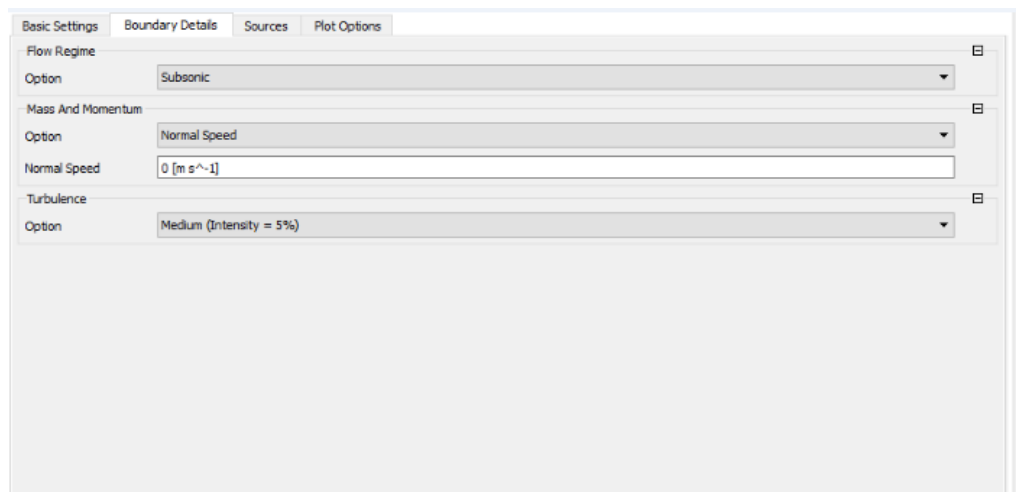


Рисунок 3.37 – Модель задание условий для входной стенки окружающего пространства исследуемой имитационной модели гексакоптера

На рисунке 3.38 представлена входная стенка окружающего пространства исследуемой имитационной модели гексакоптера.

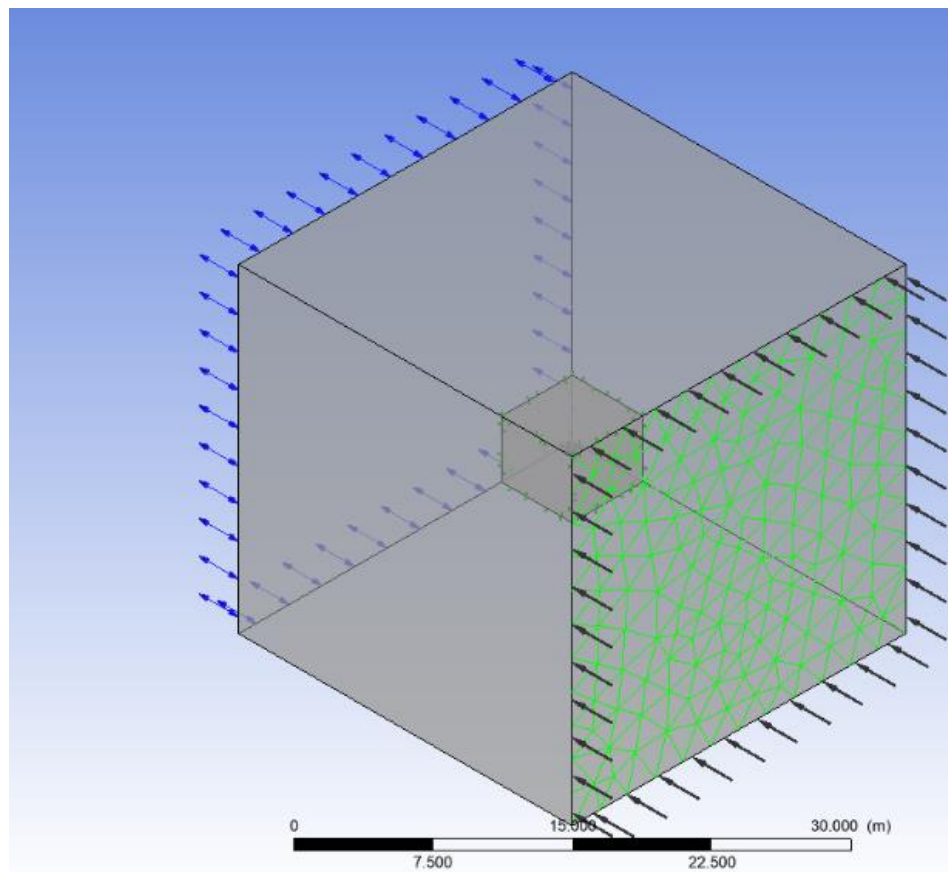


Рисунок 3.38 –Входная стенка окружающего пространства исследуемой имитационной модели гексакоптера

Результаты расчетов определения аэродинамических свойств имитационной модели гексакоптера при 0 км/ч представлены на рисунках 3.39, 3.40, 3.41, 3.42.

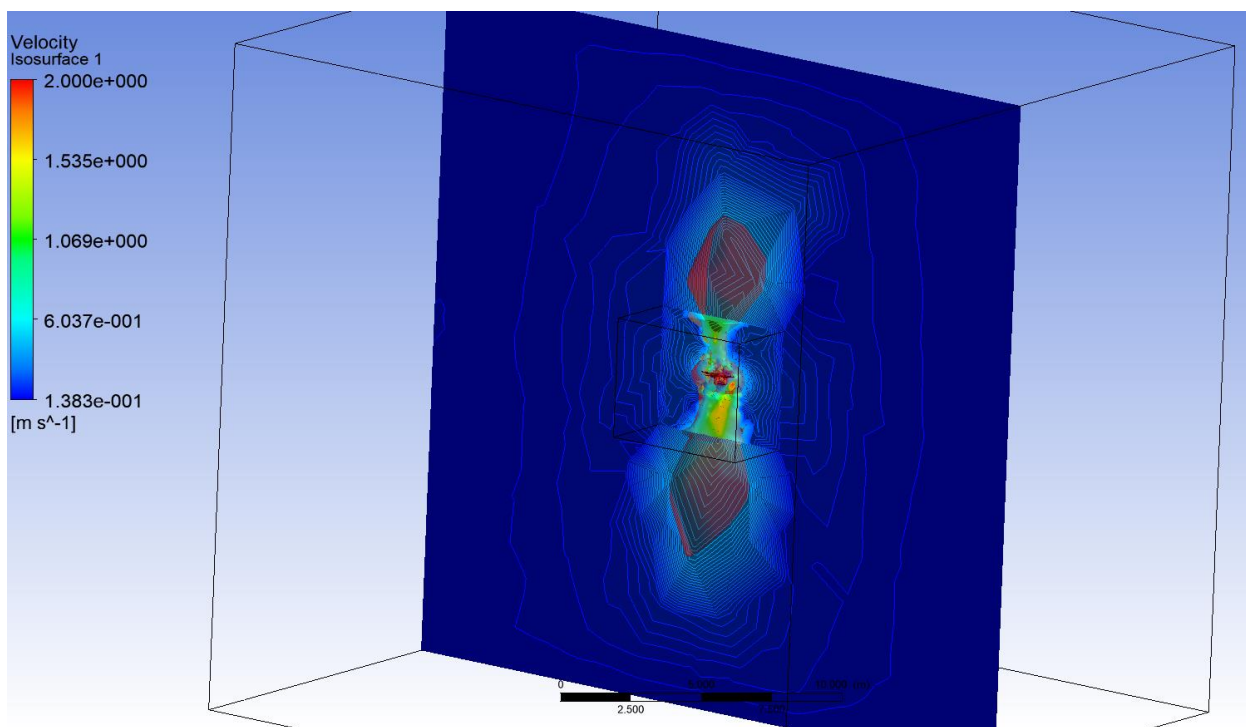


Рисунок 3.39 – Результат расчета аэродинамических свойств исследуемой имитационной модели гексакоптера при 0 км/ч

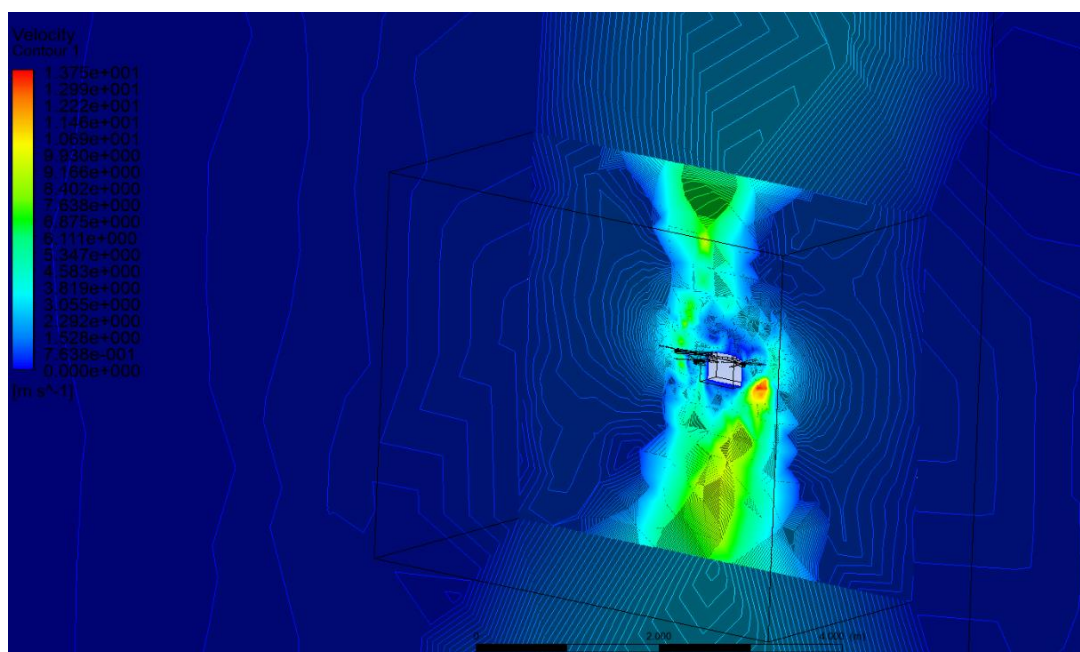


Рисунок 3.40 – Результат расчета аэродинамических свойств исследуемой имитационной модели гексакоптера при 0 км/ч, изометрия

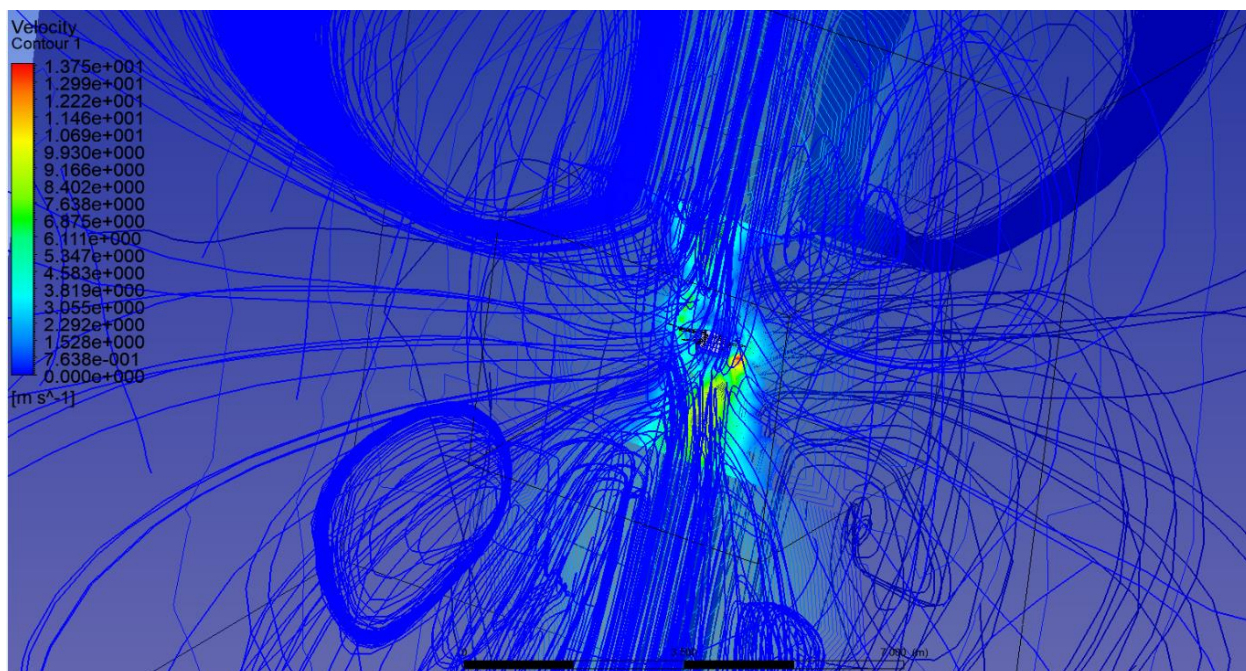


Рисунок 3.41– Результат расчета аэродинамических свойств исследуемой имитационной модели гексакоптера при 0 км/ч, вихревые потоки

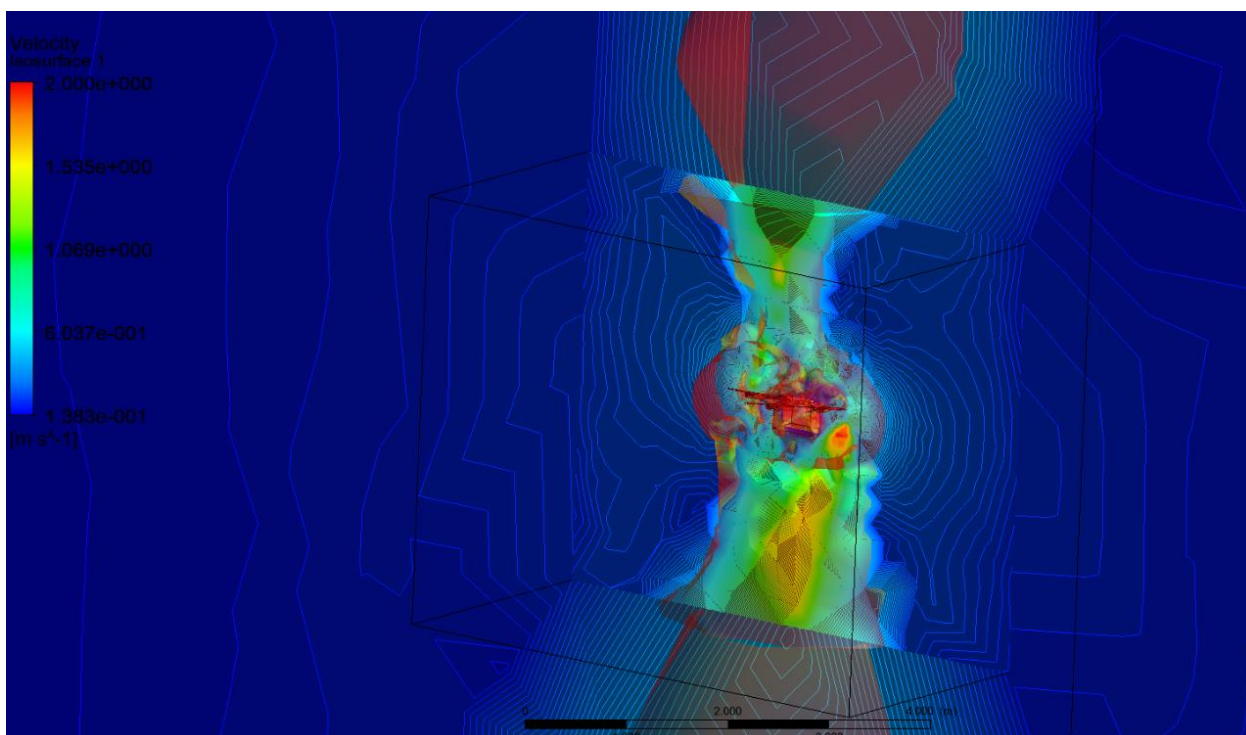


Рисунок 3.42 – Результат расчета аэродинамических свойств исследуемой имитационной модели гексакоптера при 0 км/ч, наложение скорости потока при 1 км/ч

Результаты расчетов определения аэродинамических свойств имитационной модели гексакоптера при 30 км/ч представлены на рисунках 3.43, 3.44, 3.45, 3.46.

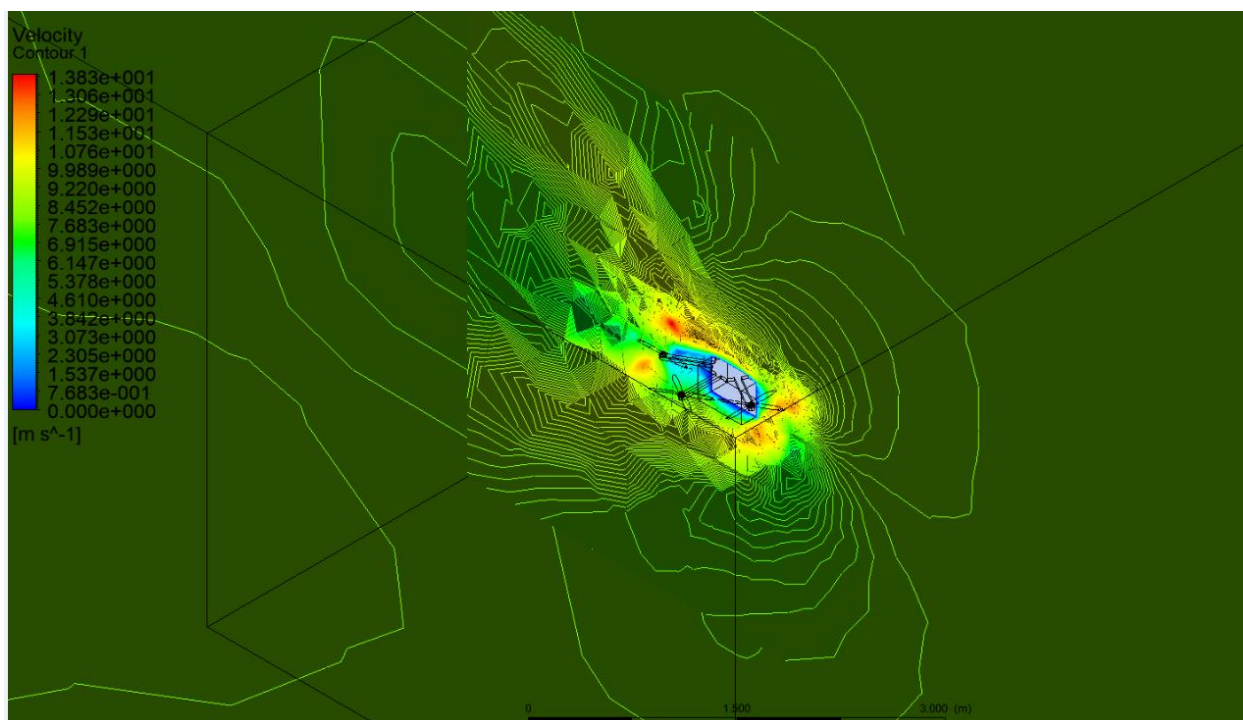


Рисунок 3.43 – Результат расчета аэродинамических свойств исследуемой имитационной модели гексакоптера при 30 км/ч

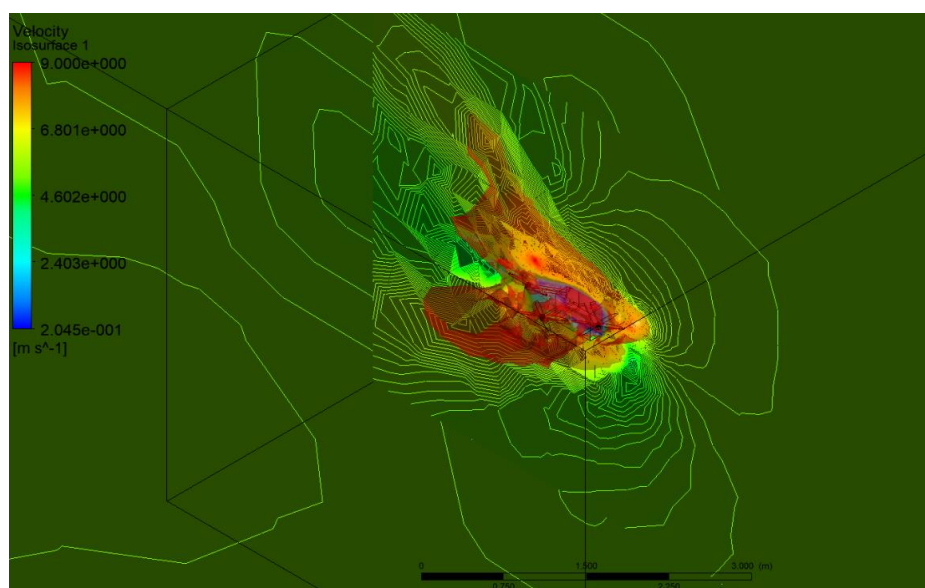


Рисунок 3.44 – Результат расчета аэродинамических свойств исследуемой имитационной модели гексакоптера при 30 км/ч, наложение скорости потока при 32 км/ч

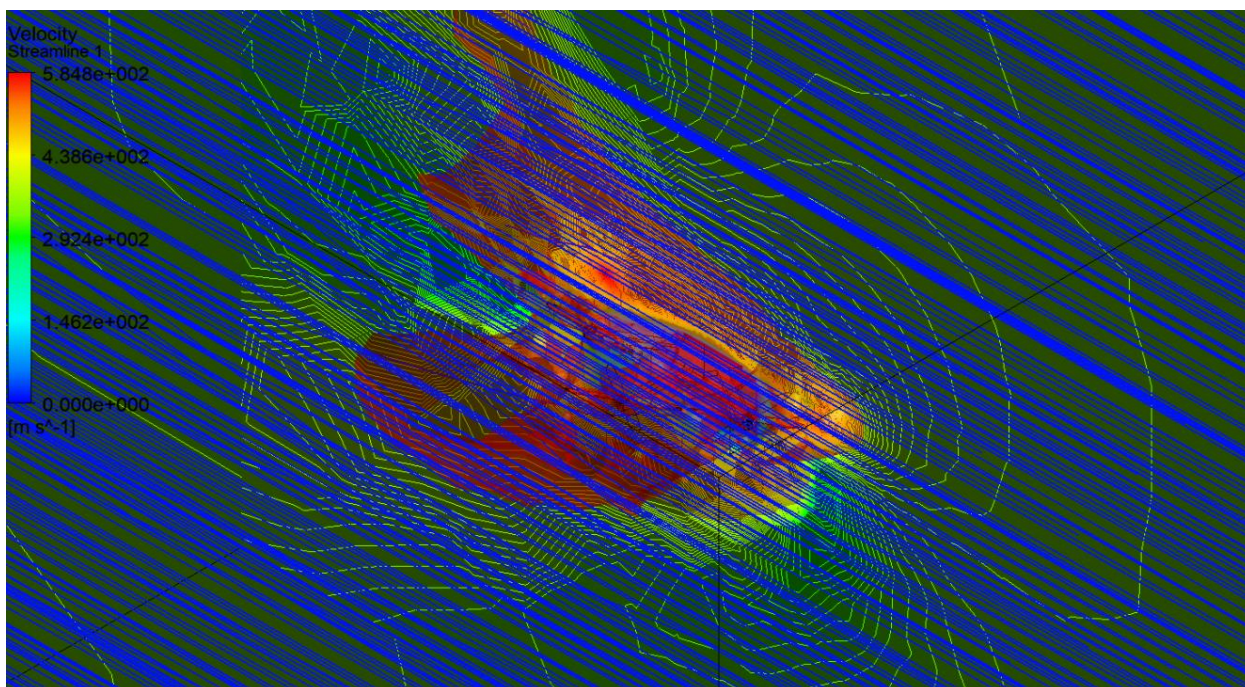


Рисунок 3.45 – Результат расчета aerодинамических свойств исследуемой имитационной модели гексакоптера при 30 км/ч, вихревые потоки и наложение скорости потока при 32 км/ч

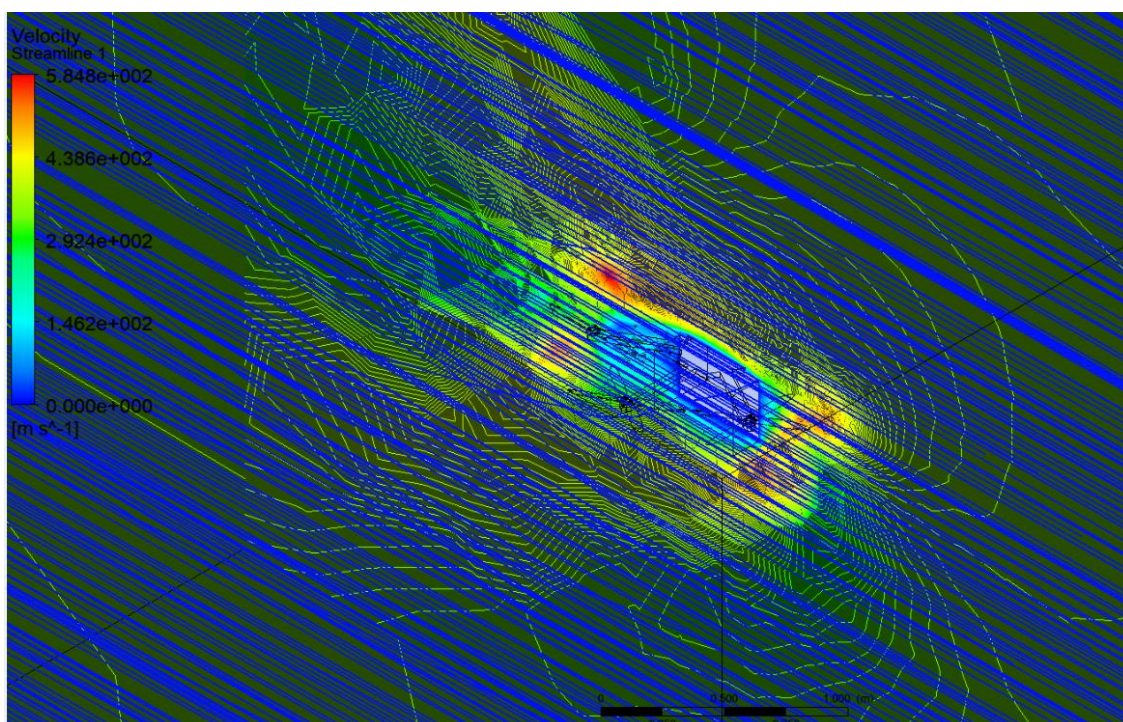


Рисунок 3.46 – Результат расчета aerодинамических свойств исследуемой имитационной модели гексакоптера при 30 км/ч, вихревые потоки

Результаты расчетов определения аэродинамических свойств имитационной модели гексакоптера при 60 км/ч представлены на рисунках 3.47, 3.48, 3.49, 3.50.

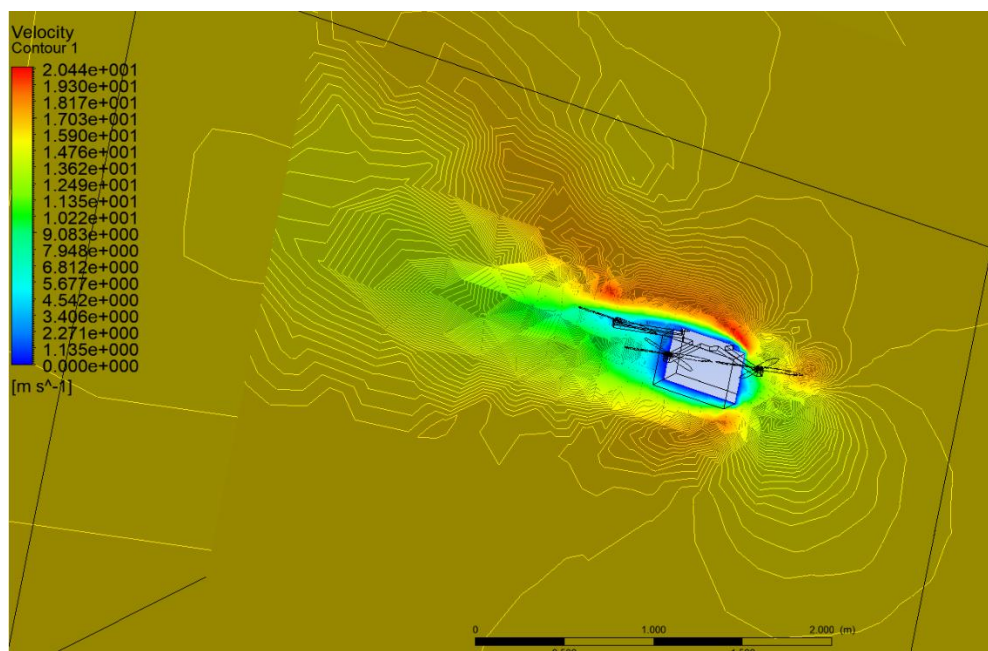


Рисунок 3.47 – Результат расчета аэродинамических свойств исследуемой имитационной модели гексакоптера при 60 км/ч

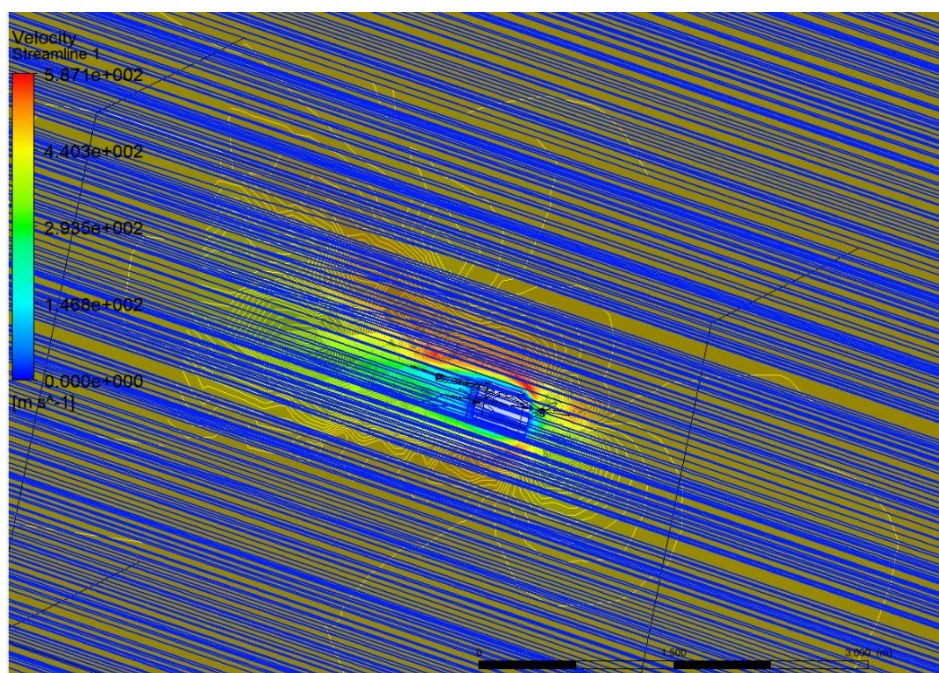


Рисунок 3.48 – Результат расчета аэродинамических свойств исследуемой имитационной модели гексакоптера при 60 км/ч, вихревые потоки

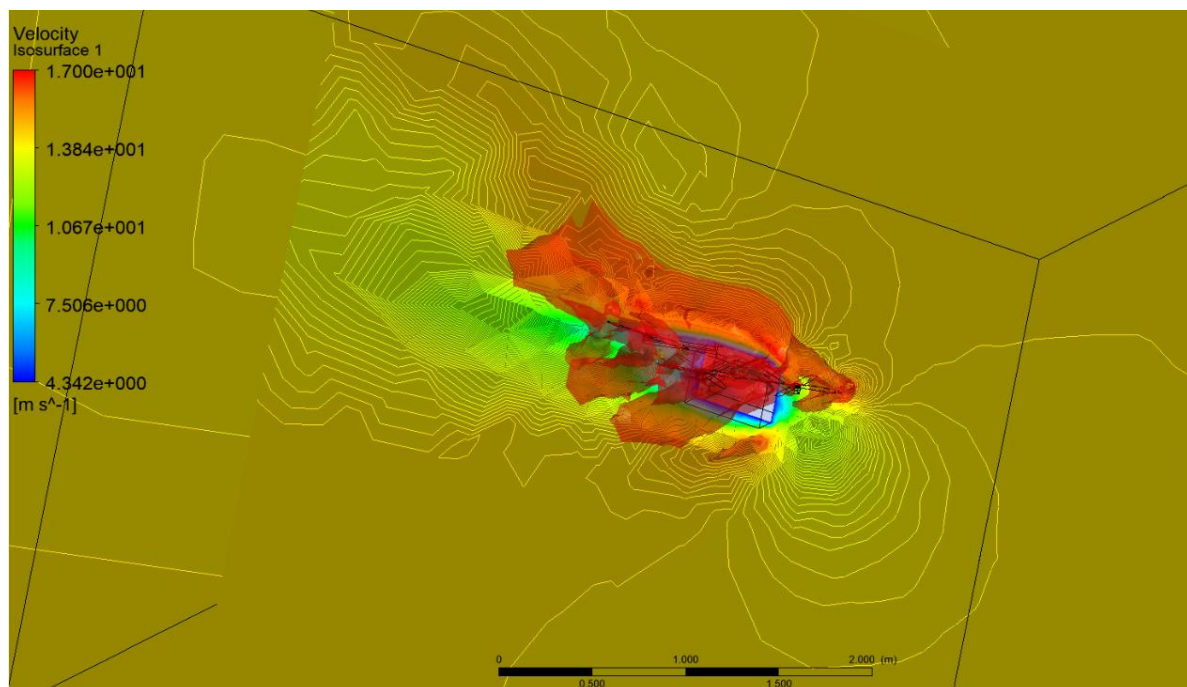


Рисунок 3.49 – Результат расчета аэродинамических свойств исследуемой имитационной модели гексакоптера при 60 км/ч, наслоение скорости потока при 62 км/ч

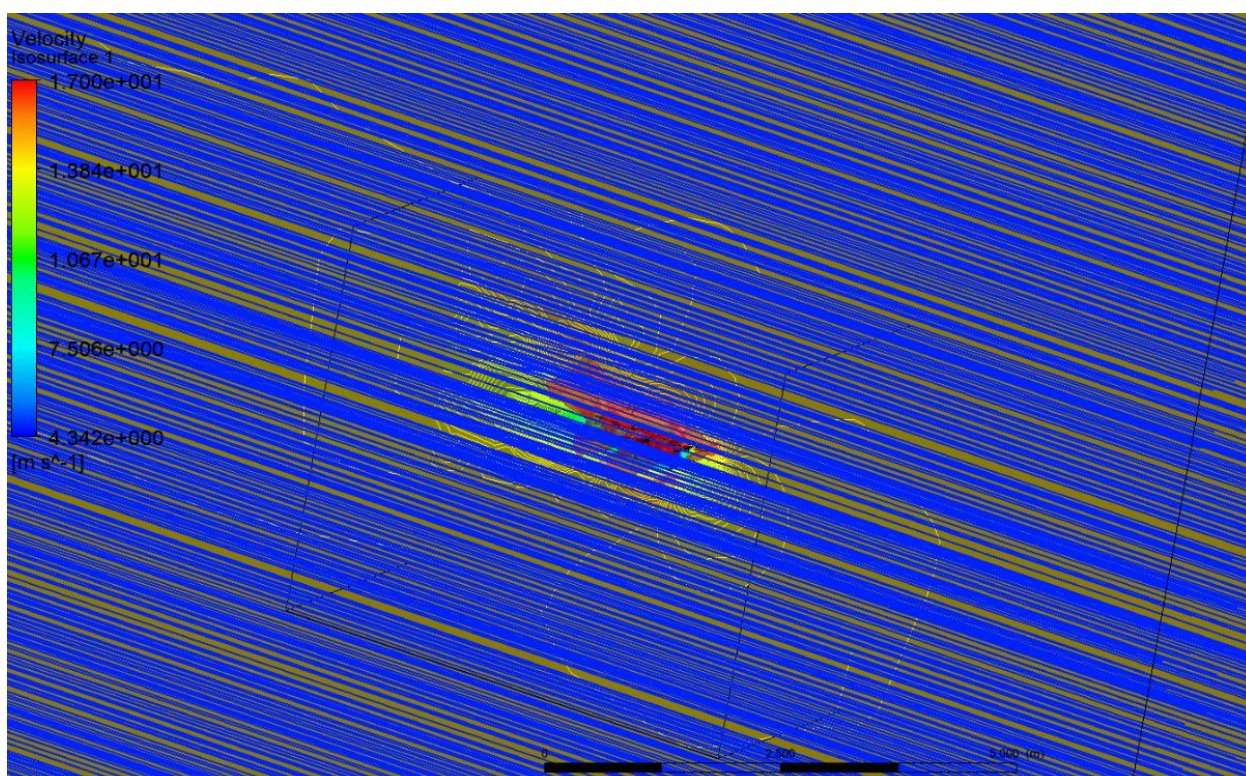


Рисунок 3.50 – Результат расчета аэродинамических свойств исследуемой имитационной модели гексакоптера при 60 км/ч, наслоение скорости потока при 62 км/ч и вихревые потоки

Результаты расчетов определения аэродинамических свойств имитационной модели гексакоптера при 100 км/ч представлены на рисунках 3.51, 3.52, 3.53, 3.54.

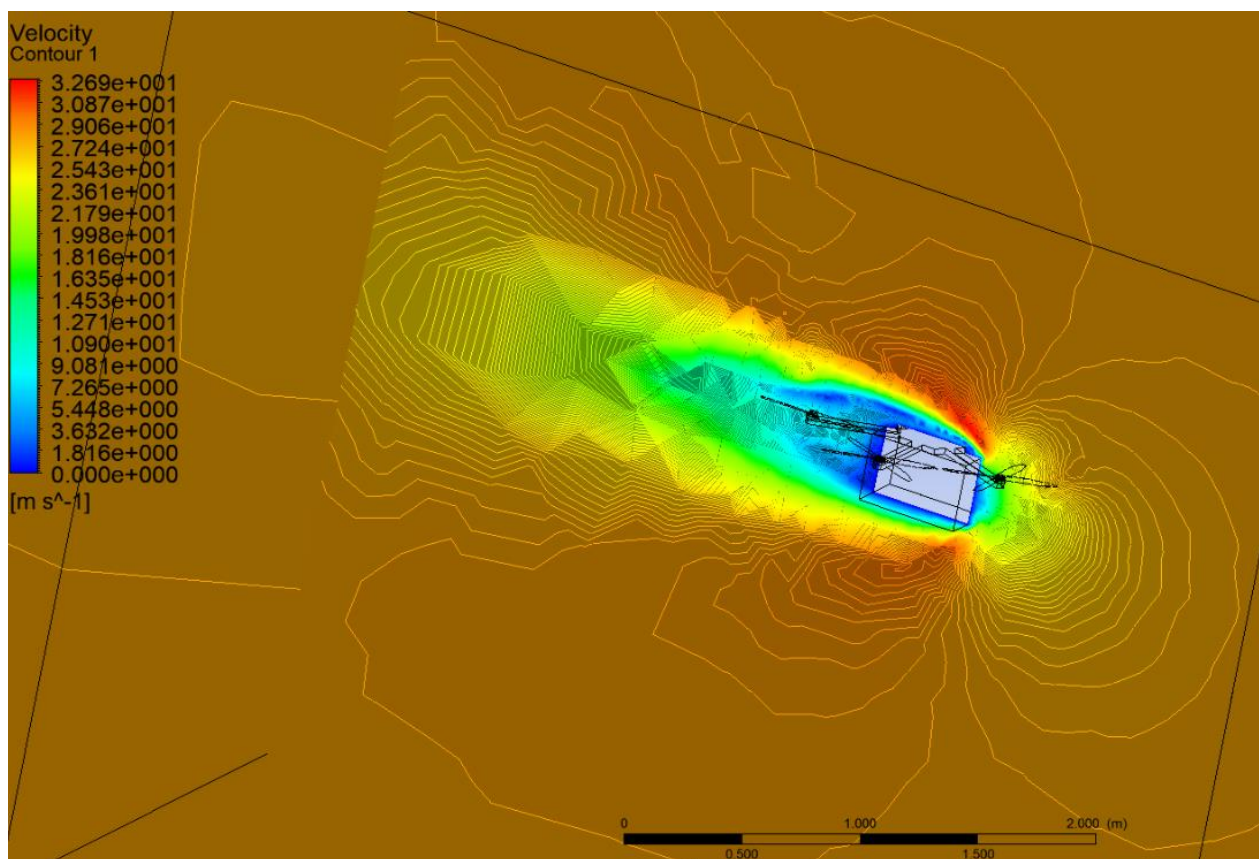


Рисунок 3.51 – Результат расчета аэродинамических свойств исследуемой имитационной модели гексакоптера при 100 км/ч

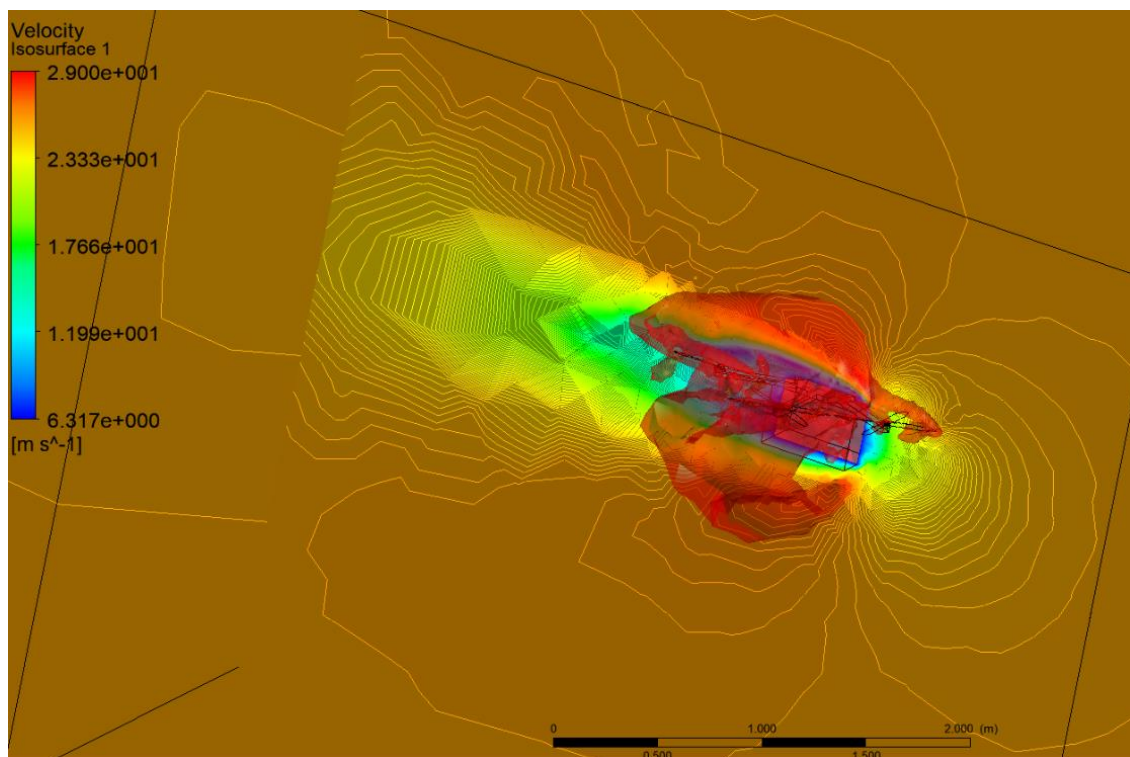


Рисунок 3.52 – Результат расчета aerодинамических свойств исследуемой имитационной модели гексакоптера при 100 км/ч, наложение скорости потока при 102 км/ч

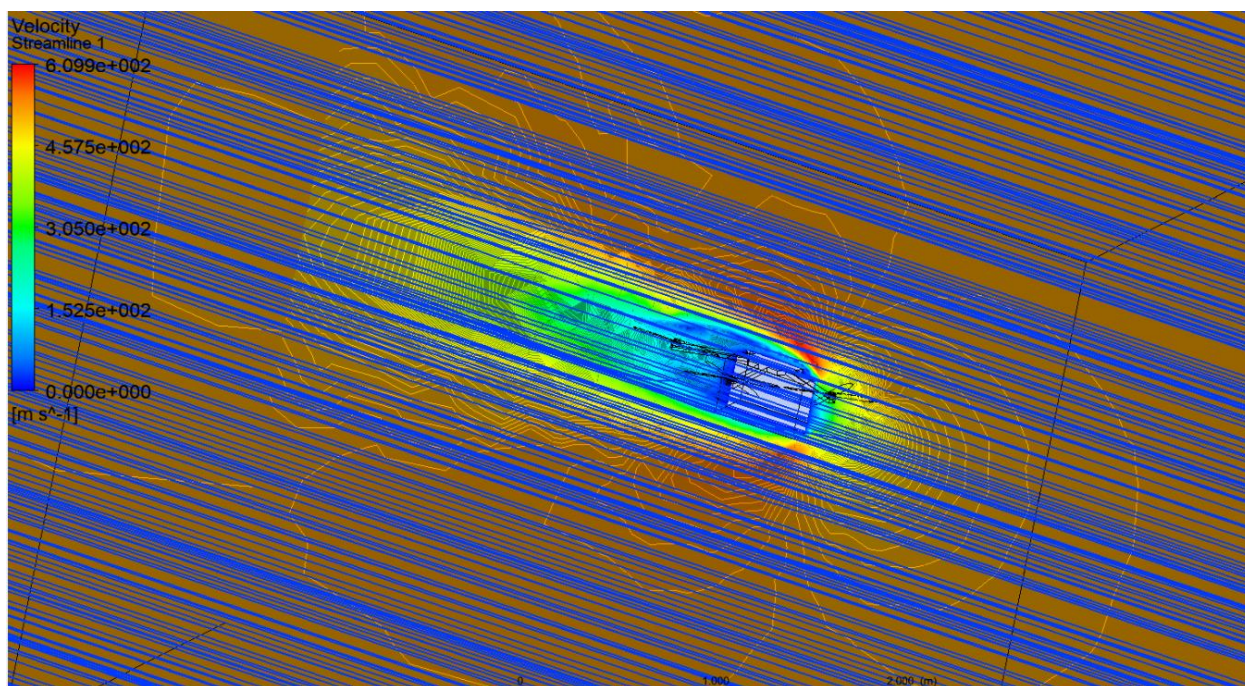


Рисунок 3.53 – Результат расчета aerодинамических свойств исследуемой имитационной модели гексакоптера при 100 км/ч, вихревые потоки

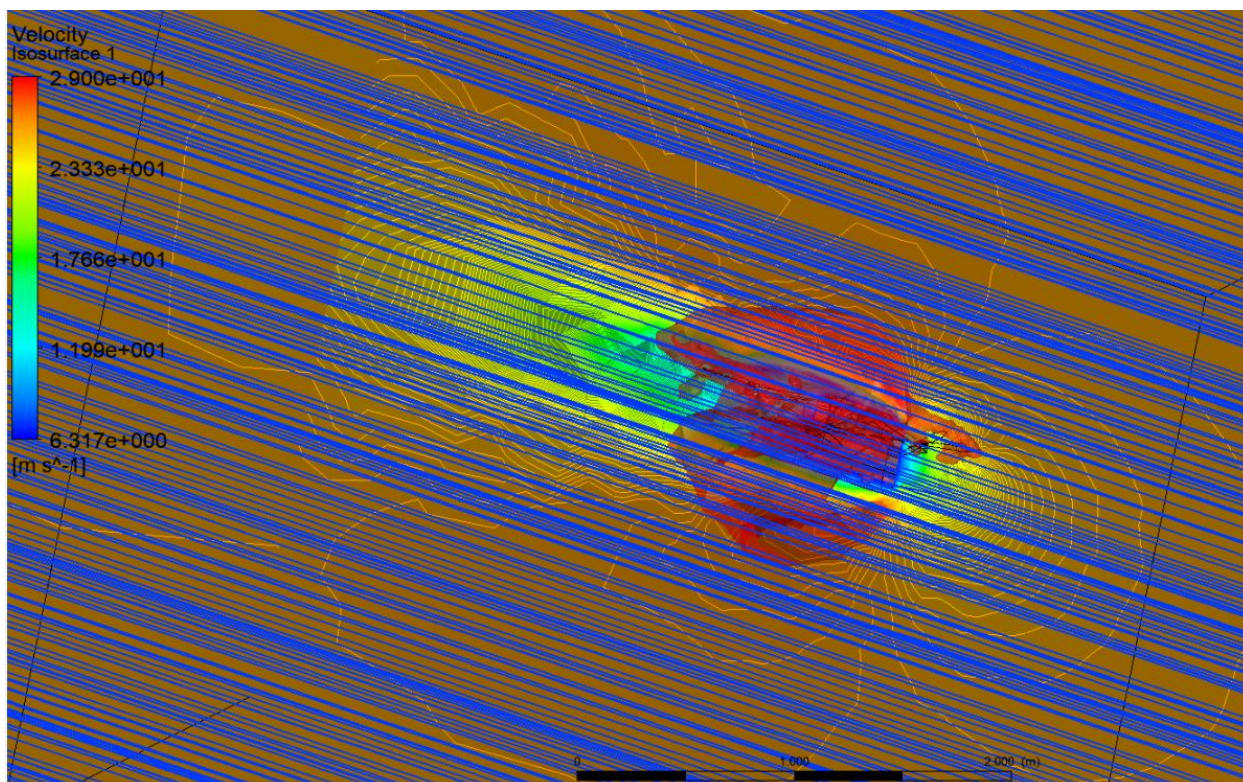


Рисунок 3.54 – Результат расчета aerодинамических свойств исследуемой имитационной модели гексакоптера при 100 км/ч, вихревые потоки и наложение скорости потока при 102 км/ч

Результаты расчетов определения aerодинамических свойств имитационной модели гексакоптера при 150 км/ч представлены на рисунках 3.55, 3.56, 3.57, 3.58.

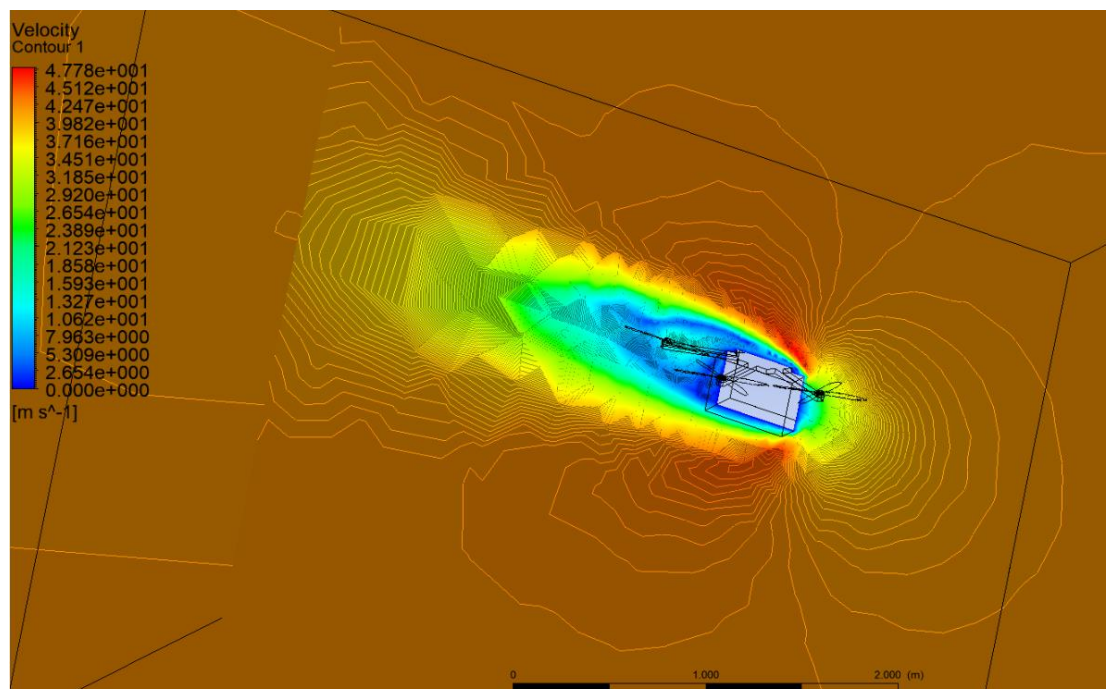


Рисунок 3.55 – Результат расчета aerодинамических свойств исследуемой имитационной модели гексакоптера при 150 км/ч

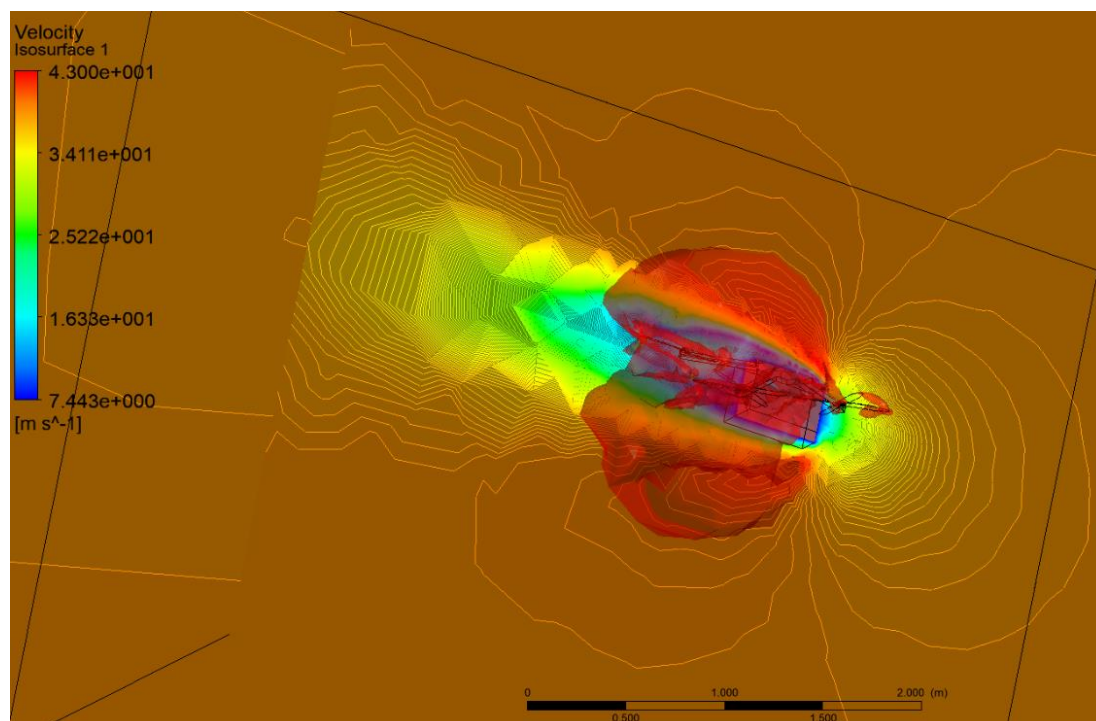


Рисунок 3.56 – Результат расчета aerодинамических свойств исследуемой имитационной модели гексакоптера при 150 км/ч, наложение скорости потока при 152 км/ч

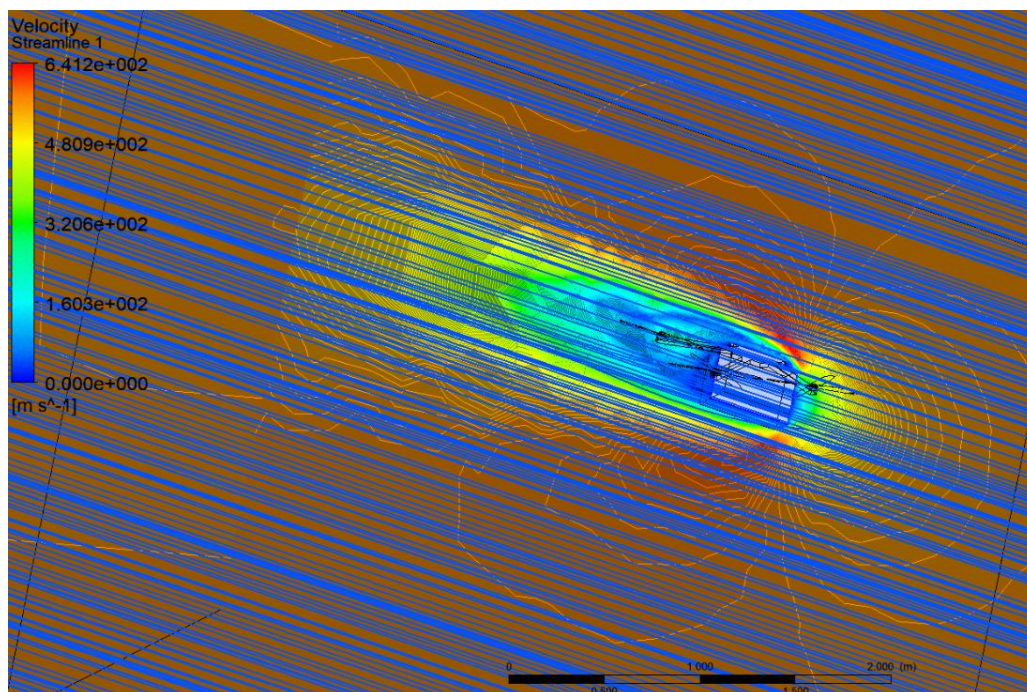


Рисунок 3.57 – Результат расчета aerодинамических свойств исследуемой имитационной модели гексакоптера при 150 км/ч, вихревые потоки

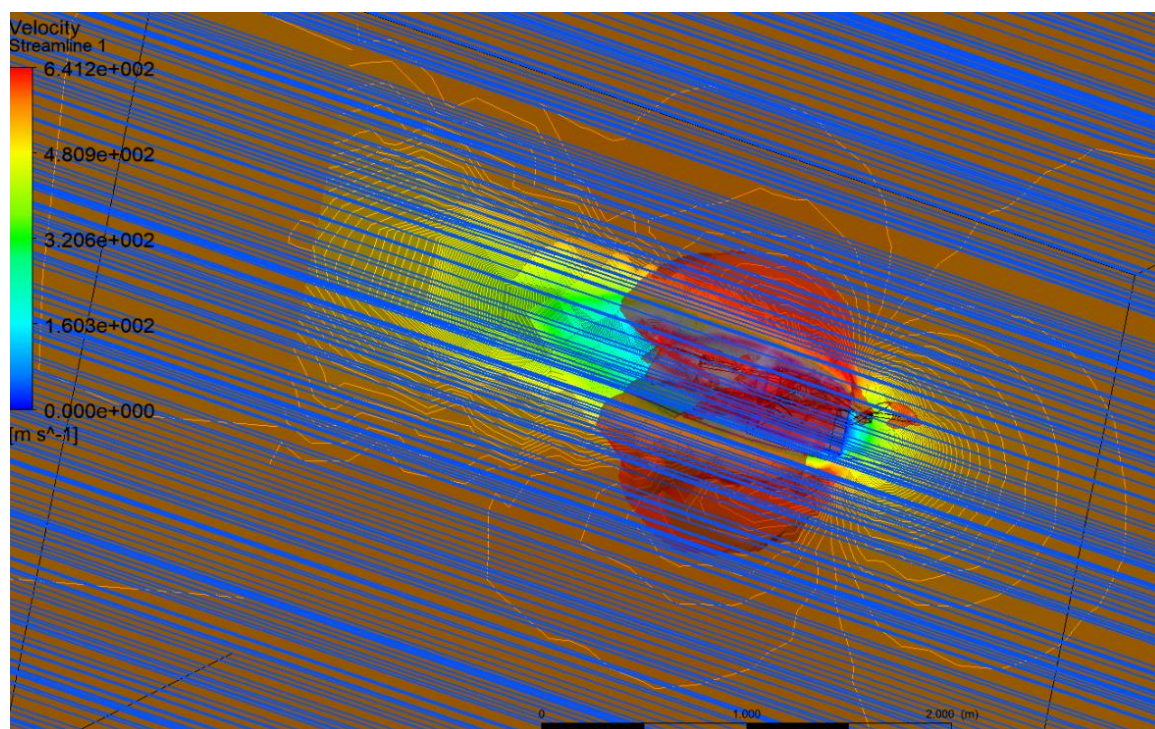


Рисунок 3.58 – Результат расчета aerодинамических свойств исследуемой имитационной модели гексакоптера при 150 км/ч, вихревые потоки и наложение скорости потока при 152 км/ч

Вывод: выполнены расчёты аэродинамических характеристик имитационной модели гексакоптера при изменении скорости бокового потока от 1 км/ч до 150 км/ч. Визуализация линий тока и поля скоростей на поверхности тела, винтах исследуемого объекта и в исследуемых областях пространства позволяет учесть особенности обтекания реального аппарата при заданных аэродинамических условиях для улучшения полета гексакоптера как с полезной нагрузкой, так и без нее.

ЗАКЛЮЧЕНИЕ

В магистерской диссертации освещены основные принципы и особенности разработки имитационных моделей для определения динамики движения БПЛА мультироторного типа. Рассмотрены основные принципы и методы определения аэродинамических и прочностных характеристик БПЛА мультироторного типа.

Проведен анализ требований к программному обеспечению для разработки имитационных моделей и расчета аэродинамических и прочностных характеристик БПЛА мультироторного типа. Изучены требования, технические условия решения задач моделирования динамики движения БПЛА мультироторного типа. Проведен аэродинамический и прочностной расчеты для БПЛА мультироторного типа.

Разработана имитационная модель для изучения динамики движения мультикоптера. По результатам работы модели удалось создать сценарий полета гексакоптера по точкам миссии с использованием технологий лидара и камеры. Был подготовлен и протестирован сценарий с использованием алгоритма Sense and Avoid. Алгоритм позволяет гексакоптеру с использованием датчиков, таких как лидар и камера, облетать препятствия на проложенном пути. Также был проведен расчет аэродинамических и прочностных характеристик имитационной модели гексакоптера при изменении скорости бокового потока от 1 км/ч до 150 км/ч. Визуализация линий тока и поля скоростей на поверхности тела, винтах исследуемого объекта и в исследуемых областях пространства позволяет учесть особенности обтекания реального аппарата при заданных аэродинамических условиях для улучшения полета гексакоптера как с полезной нагрузкой, так и без нее.

Для разработки имитационной модели динамики движения мультикоптера использовался программный пакет Gazebo. Для расчета

прочностных и аэродинамических характеристик имитационной модели использовался программный пакет ANSYS Workbench 18.2.

Реализованная в работе имитационная модель динамики движения мультикоптера, может применяться для решения практических задач динамики движения реальных объектов, так как позволяет создавать достаточно реалистичные сценарии для решения конкретных задач. Проведенный расчет аэродинамических и прочностных характеристик имитационной модели позволяет достаточно точно определить данные характеристики реальной модели, схожей по конструкции с имитационной моделью.

Результаты работы опубликованы в [29-33].

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Autodesk Inventor. [Электронный ресурс]. – Режим доступа: https://ru.wikipedia.org/wiki/Autodesk_Inventor.
2. SolidWorks. [Электронный ресурс]. – Режим доступа: <http://www.edgecamsoftware.ru/cam-for-solidworks>.
3. Обзор программы SolidWorks. [Электронный ресурс]. – Режим доступа: <https://3ddevice.com.ua/blog/3d-printer-obzor/obzor-programmy-solidworks/>.
4. Ansys mechanical interface. [Электронный ресурс]. – Режим доступа: <http://3dtoday.ru/blogs/dagov/ansys-mechanical-interface/>.
5. Mesh-сетка. [Электронный ресурс]. – Режим доступа: <http://www.k204.ru/books/meshes.pdf>.
6. Динамика жидкостей и газов. [Электронный ресурс]. – Режим доступа: <https://cae-expert.ru/discipline/dinamika-zhidkostey-i-gazov>.
7. ИПИ-ТЕХНОЛОГИИ В ПРИБОРОСТРОЕНИИ: УЧЕБНОЕ ПОСОБИЕ. [Электронный ресурс]. – Режим доступа: http://window.edu.ru/catalog/pdf2txt/743/58743/28643?p_page=4.
8. Использование 3D моделей на различных этапах ЖЦИ. [Электронный ресурс]. – Режим доступа: https://studopedia.su/10_123752_ispolzovanie-D-modeley-na-razlichnih-etapah-zhtsi.html.
9. Программы для 3D моделирования. [Электронный ресурс]. – Режим доступа: <https://freelance.today/poleznoe/20-besplatnyh-programm-dlya-3d-modelirovaniya.html>.
10. ANSYS CFD 14.5. [Электронный ресурс]. – Режим доступа: <http://www.sccc.icmmg.nsc.ru/asoftware.html>.

11. Исследование аэродинамических характеристик профиля и законцовок лопасти несущего винта вертолѐта методами CFD. [Электронный ресурс]. – Режим доступа: <https://docplayer.ru/31526352>
12. Полетные режимы PX4. [Электронный ресурс]. – URL: <https://clever.copterexpress.com/ru/modes.html>
13. Следящая система. [Электронный ресурс]. – URL: <https://www.booksite.ru/fulltext/1/001/008/103/202.htm> (дата обращения: 26.03.2019).
14. Параметры PX4. [Электронный ресурс]. – URL: https://clever.copterexpress.com/ru/px4_parameters.html
15. Поворотная антенна. [Электронный ресурс]. – URL: http://www.parkflyer.ru/ru/blogs/view_entry/12055/
16. Mission Planner Home. [Электронный ресурс]. – URL: <http://ardupilot.org/planner/>
17. Raspberry Pi 3 Model B. [Электронный ресурс]. – URL: <https://www.raspberrypi.org/products/raspberry-pi-3-model-b/>
18. Pixhawk Series. [Электронный ресурс]. – URL: https://docs.px4.io/en/flight_controller/pixhawk_series.html
19. Pixhawk Autopilot. [Электронный ресурс]. – URL: <https://pixhawk.org/modules/pixhawk>
20. PX4 Basic Concepts. [Электронный ресурс]. – URL: https://docs.px4.io/en/getting_started/px4_basic_concepts.html
21. Robot Operating System. [Электронный ресурс]. – URL: <http://wiki.ros.org/ROS/Introduction>
22. Why ROS 2.0? [Электронный ресурс]. – URL: http://design.ros2.org/articles/why_ros2.html (дата обращения: 15.05.2019).
23. MAVROS. [Электронный ресурс]. – URL: <http://wiki.ros.org/mavros>
24. MAVLink Micro Air Vehicle Communication Protocol. [Электронный ресурс]. – URL: <http://qgroundcontrol.org/mavlink/start>
25. PX4. [Электронный ресурс]. – URL: <https://px4.io>

26. PX4 Autopilot Software. [Электронный ресурс]. – URL: <https://github.com/PX4/Firmware>
27. An Autonomous Vehicle Navigation Algorithm. [Электронный ресурс]. – URL: https://www.researchgate.net/publication/271541690_An_Autonomous_Vehicle_Navigation_Algorithm
28. Implementation of autonomous vehicle navigation algorithms using event-driven programming. [Электронный ресурс]. – URL: <https://ieeexplore.ieee.org/document/6306150>
29. II Международная конференция «Когнитивная Робототехника» в рамках Международного форума «Интеллектуальные системы 4-й промышленной революции» (1) (22.11.17-25.11.17). А. А. Таганов – Инерциальная навигационная система для беспилотного летательного аппарата – Когнитивная робототехника: материалы II международной конференции (22-25 ноября 2017 г.) / под ред. В.И. Сырымкина, А.В. Юрченко; Национальный исследовательский Томский государственный университет – 86 с.
30. II Международная конференция «Когнитивная Робототехника» в рамках Международного форума «Интеллектуальные системы 4-й промышленной революции» (1) (22.11.17-25.11.17). А. А. Таганов – Моделирование режимов полета беспилотного летательного аппарата – Когнитивная робототехника: материалы II международной конференции (22-25 ноября 2017 г.) / под ред. В.И. Сырымкина, А.В. Юрченко; Национальный исследовательский Томский государственный университет – 86 с.
31. Таганов А. А., Шидловский С. В. - Моделирование и тестирование динамики движения компьютерной твердотельной модели беспилотного летательного аппарата. опτικο-электронные приборы и устройства в системах распознавания образов, обработки изображений и символьной информации. распознавание - 2018

Сборник материалов XIV международной научно-технической конференции. 2018

Издательство: Юго-Западный государственный университет (Курск), С 255-256

32. A. A. Taganov, D. V. Shashev. Simulation of the behavior of an unmanned aerial vehicle in virtual 3D scenes. IOP Conf. Ser.: Mater. Sci. Eng. 516 012046

33. XIV Международная школа-конференция студентов, аспирантов, молодых ученых «Инноватика-2018». Таганов А.А. - Моделирование динамики движения компьютерной твердотельной модели беспилотного летательного аппарата - Инноватика-2018: сб. материалов XIV Международной школы-конференции студентов, аспирантов и молодых ученых (26–27 апреля 2018 г.) / под ред. А.Н. Солдатов, С.Л. Минькова. – Томск: STT, 2018. – 556 с.



АНТИПЛАГИАТ
ТВОРИТЕ СОБСТВЕННЫМ УМОМ

Томский Государственный
Университет

СПРАВКА

о результатах проверки текстового документа
на наличие заимствований

Проверка выполнена в системе
Антиплагиат.ВУЗ

Автор работы	Таганов Александр Александрович
Подразделение	ФИТ, кафедра информационного обеспечения инновационной деятельности
Тип работы	Магистерская диссертация
Название работы	ИМИТАЦИОННОЕ МОДЕЛИРОВАНИЕ ДИНАМИКИ ДВИЖЕНИЯ МУЛЬТИКОПТЕРА
Название файла	ВКР КВАДРОТОРЫ.pdf
Процент заимствования	9,23%
Процент цитирования	6,70%
Процент оригинальности	84,07%
Дата проверки	05:43:22 20 июня 2019г.
Модули поиска	Сводная коллекция ЭБС; Модуль поиска ИПС "Адилет"; Модуль поиска общеупотребительных выражений; Кольцо вузов; Модуль поиска "ТГУ"; Коллекция Патенты; Модуль поиска перефразирований Интернет; Модуль поиска перефразирований eLIBRARY.RU; Коллекция Медицина; Модуль поиска Интернет; Коллекция ГАРАНТ; Коллекция eLIBRARY.RU; Модуль поиска переводных заимствований; Цитирование; Коллекция РГБ; Модуль выделения библиографических записей
Работу проверил	Гольцова Полина Андреевна ФИО проверяющего
Дата подписи	20.06.19. Подпись проверяющего

Чтобы убедиться
в подлинности справки,
используйте QR-код, который
содержит ссылку на отчет.



Ответ на вопрос, является ли обнаруженное заимствование
корректным, система оставляет на усмотрение проверяющего.
Предоставленная информация не подлежит использованию
в коммерческих целях.