

УДК 004.588
DOI 10.17513/snt.40167

РАЗРАБОТКА СИМУЛЯТОРА ДЛЯ ОБУЧЕНИЯ ОПЕРАТОРОВ БЕСПИЛОТНЫХ ЛЕТАТЕЛЬНЫХ АППАРАТОВ

Галкин Д.В., Петухов И.В., Танрывердиев И.О., Стешина Л.А.,
Стешин И.С., Курасов П.А.

ФГБОУ ВО «Поволжский государственный технологический университет», Йошкар-Ола,
e-mail: GalkinDV@volgatech.net, rector@volgatech.net, TanryverdievIO@volgatech.net,
SteshinaLA@volgatech.net, SteshinaIS@volgatech.net, KurasovPA@volgatech.net

Данная статья посвящена созданию симулятора для обучения операторов беспилотных летательных аппаратов с использованием среды разработки Unity. В работе подробно описан процесс создания симулятора, начиная с выбора подходящей платформы благодаря её богатой системе плагинов и моделей. Особое внимание уделено интеграции специализированного контроллера Radiomaster TX12 через плагин Input System, что позволяет определять используемые сигналы с контроллера и настраивать переменные для управления. В статье изложены методы применения ПИД-регуляторов для более реалистичной стабилизации беспилотных летательных аппаратов в симуляторах. Рассмотрены функции для расчёта ошибок по осям наклона и вращения, а также нормализации углов, обеспечивающие корректное управление движением. Также с целью реализации вида от первого лица был создан скрипт для камеры, фиксирующий её на беспилотном летательном аппарате и стабилизирующий тряску камеры при полёте. Обсуждаются этапы написания кода, включая объявление переменных и подключение контроллера для чтения данных. Предоставляется подробный обзор функций программы, отвечающих за физику поведения дрона и управление с помощью контроллера. Данный симулятор позволяет моделировать реальное управление беспилотным летательным аппаратом, представляя его как ценное средство для обучения и развития навыков оператора.

Ключевые слова: беспилотные летательные аппараты, симулятор, обучение, Unity, моделирование, обучение операторов, управление, визуализация, ПИД-регулятор

Исследование выполнено при поддержке гранта Российского научного фонда № 23-19-00568 «Методы и интеллектуальная система поддержки динамической устойчивости операторов эргатических систем», <https://rscf.ru/project/23-19-00568/>.

DEVELOPMENT OF A SIMULATOR FOR TRAINING DRONE OPERATORS

Galkin D.V., Petukhov I.V., Tanryverdiev I.O., Steshina L.A.,
Steshin I.S., Kurasov P.A.

Volga State University of Technology, VSUT, Volgatech, Joshkar-Ola,
e-mail: rector@volgatech.net, TanryverdievIO@volgatech.net, SteshinaLA@volgatech.net,
SteshinaIS@volgatech.net, KurasovPA@volgatech.net, GalkinDV@volgatech.net

This paper is devoted to the creation of a simulator for training drone operators using the Unity development environment. The paper describes in detail the process of creating the simulator, starting with the selection of a suitable platform due to its rich system of plug-ins and models. Particular attention is paid to the integration of a specialized Radiomaster TX12 controller through the “Input System” plugin, which allows to define the signals used from the controller and to set up variables for control. The article describes methods of using PID controllers for more realistic stabilization of unmanned aerial vehicles in simulators. Functions for calculating errors in the axes of tilt and rotation, as well as normalization of angles, providing correct motion control, are considered. Also, in order to implement the first-person view, a script for the camera fixing it on the drone and stabilizing the camera shake during flight was created. The steps involved in writing the code are discussed, including declaring variables and connecting a controller to read the data. A detailed overview of the program functions responsible for the physics of drone behavior and controller control is provided. This simulator allows you to model real drone control, presenting it as a valuable tool for learning and developing operator skills.

Keywords: unmanned aerial vehicles, simulator, training, Unity, simulation, operator training, control, visualization, PID controller

The study was supported by the Russian Science Foundation grant No. 23-19-00568 “Methods and intelligent system for supporting dynamic stability of operators of ergatic systems”, <https://rscf.ru/project/23-19-00568/>.

Введение

Создание симулятора для обучения операторов беспилотных летательных аппаратов (БПЛА) является важной инициативой в условиях растущего использования дро-

нов в различных отраслях, таких как сельское хозяйство, строительство и экстренные службы. Симулятор предоставляет безопасную и контролируемую среду для обучения, минимизируя риски и сокращая затраты

на обучение [1; 2]. Он позволяет отрабатывать действия в аварийных ситуациях и адаптировать сценарии под специфические нужды, что способствует эффективной подготовке операторов к реальным условиям полета. Кроме того, использование современных технологий обеспечивает высокую визуализацию и реалистичную физику, что делает обучение более интерактивным и вовлекающим [3; 4].

Такой подход не только развивает навыки новых операторов, но и повышает квалификацию опытных специалистов, готовых справляться с разнообразными задачами, такими как инспекции и спасательные операции. Таким образом, симулятор становится важным инструментом для повышения безопасности и эффективности эксплуатации БПЛА.

Цель исследования – разработка симулятора для обучения операторов беспилотных летательных аппаратов, который обеспечит их высококачественную подготовку.

Материалы и методы исследования

В этом исследовании для разработки симулятора был использован программный продукт Unity, предоставляющий мощные инструменты для создания среды виртуальной реальности и моделирования физических процессов. Unity позволил не только разработать визуальную среду, но и интегрировать различные плагины и сценарии, необходимые для функционирования симулятора управления БПЛА. В частности, был добавлен плагин Input System, обеспечивающий взаимодействие с контроллерами и позволяющий настраивать считывание входных сигналов.

Для управления БПЛА в симуляторе был использован PID-контроллер, который обеспечивает стабилизацию летательного аппарата. Этот контроллер был интегрирован в проект через скрипт, написанный на языке C#.

Для полноценного тренинга были настроены сигналы с контроллера Radiomaster TX12, который используется для передачи команд управления в симулятор. Это включало определение положений джойстиков и их обработку для нормализации и инверсии сигналов.

Кроме того, были разработаны функции, отвечающие за корректное управление параметрами движения и стабилизации аппарата в виртуальной среде. Это включало расчет ошибок по осям наклона и вращения, нормализацию углов и стабилизацию камеры для уменьшения тряски.

Настройка и тестирование различных сценариев полета проводились в специаль-

но созданной среде в Unity, что позволило воспроизвести условия и задачи, с которыми могут столкнуться операторы БПЛА.

Результаты исследования и их обсуждение

В данной работе использован программный продукт Unity, позволяющий разрабатывать среду погружения с возможностью использования множества плагинов, руководств, которые помогают при разработке подобных программ [5], упрощающих процесс разработки, своя библиотека пользовательских ресурсов, из которых можно взять готовые модели, текстуры и многое другое, и также она предоставляет мощные инструменты для моделирования физических процессов.

После определения среды разработки необходимо найти модель БПЛА и импортировать её в Unity. Это всё делается простым поиском в интернете либо во внутреннем магазине Unity. Этот процесс довольно прост, и поскольку симулятор предполагает вид от первого лица, в котором БПЛА видно не будет, в данной статье внимание на этом не будет акцентировано. По итогу создания нового проекта в Unity и импортирования в него модели БПЛА, проект будет иметь вид, который представлен на рисунке 1.

Далее, поскольку симулятор предполагает использование специализированного контроллера Radiomaster TX12, нужно прописать в Unity, какие данные от контроллера будут использоваться. Для этого в Unity существует специальный плагин Input System, который добавляют через менеджер пакетов Unity.

После установки данного плагина в проекте появляется возможность создания нового типа файла Input actions, в нем и будет прописано, какие сигналы с контроллера будут использоваться и с каких переменных их можно будет считать.

Открыв данный файл, в нем необходимо создать новую карту действий, и поскольку для данного симулятора понадобятся сигналы с левого и правого джойстиков контроллера, у которых необходимо фиксировать положение по осям X и Y, понадобится создать 4 переменные и указать, какой сигнал с контроллера они советуют.

Также в этом окне есть возможность предварительно обработать данные, которые будут поступать в переменные, и для простоты дальнейшей работы сигнал с правого джойстика по оси Y был инвертирован, а сигнал с левого джойстика по оси Y был нормирован от -1 до 1, поскольку изначально он изменялся от 0 до 2.

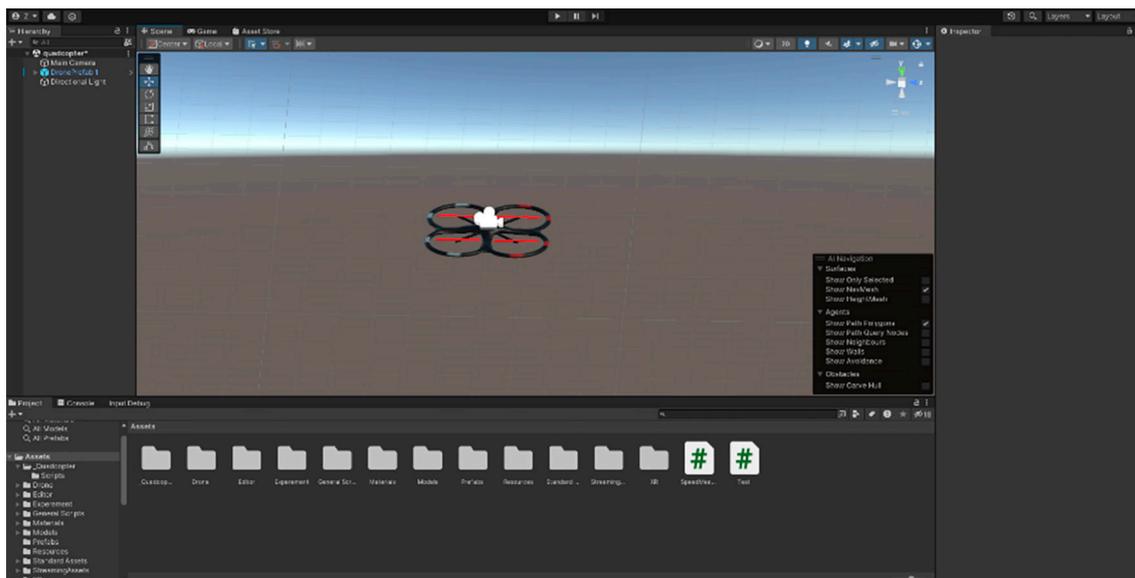


Рис. 1. Проект в Unity после добавления модели БПЛА

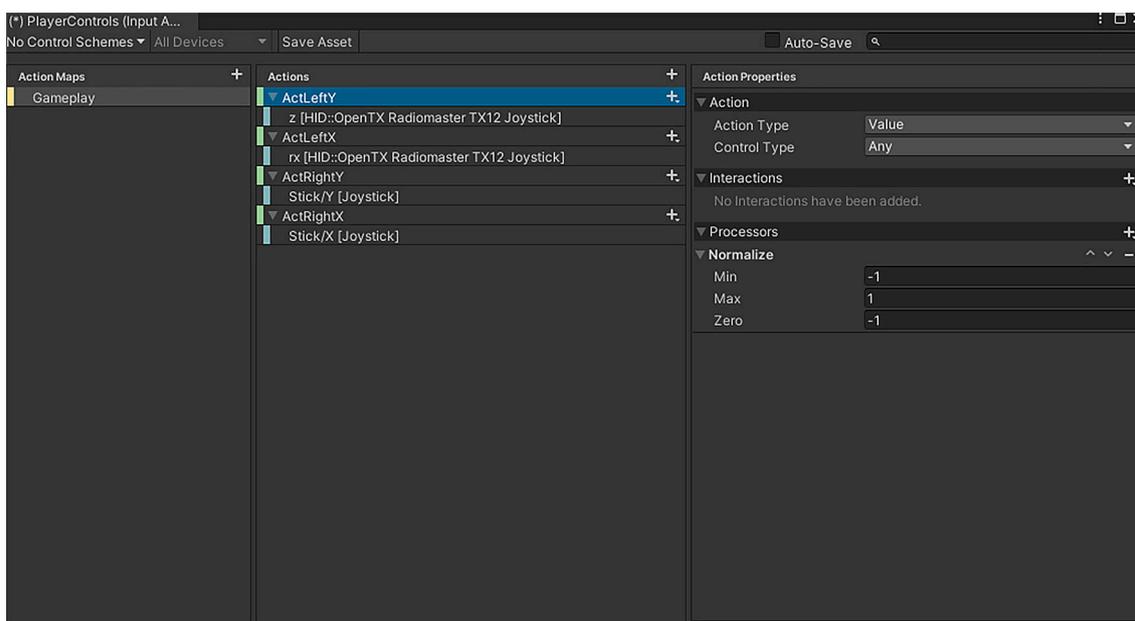


Рис. 2. Окно программы с добавленными переменными

Важным моментом при создании данных переменных является то, что нужно заранее подключить контроллер к компьютеру, так как без этого невозможно будет указать действия контроллера, к которым привязаны данные переменные.

На рисунке 2 представлен итог добавления этих переменных.

После сохранения изменений в данном файле ещё одним важным моментом является установка галочки **Generate C# Class**, она автоматически сгенерирует файл с указанными ранее переменными на язык

С#, через который и будет организовано управление.

После завершения всех приготовлений можно перейти к написанию программы, которая бы позволила управлять БПЛА.

Дальше по тексту рассмотрены основные части данного скрипта. В начале, как и в большинстве программ, в данном коде указываются именно те пространства, которые будут в нем использоваться, это позволяет использовать классы и функции без необходимости указывать полное имя каждого из них.

Далее идет часть, в которой объявляются переменные, включая RigidBody и параметры для пропеллеров, позволяющие управлять физикой БПЛА и настраивать его поведение.

Далее идет функция Awake, которая вызывается самой первой при запуске программы. В ней подключается ПИД контроллер и считываются данные о положении джойстиков контроллера.

После рассмотрена функция FixedUpdate, в ней осуществляется управление физикой БПЛА, её можно назвать основной для данной программы.

Идущие следом функции OnEnable и OnDisable нужны исключительно для считывания данных с контроллера, они начинают и завершают его.

Далее идет функция AddControls, преобразующая данные, получаемые с контроллера, в параметры движения и наклона.

После рассмотрены три функции, необходимые для стабилизации БПЛА при помощи ПИД-регуляции, что важно для более реалистичного управления движением в виртуальной реальности [6; 7]. Это функции:

GetPitchError – вычисляет ошибку по оси наклона;

GetRollError – вычисляет ошибку по оси вращения;

WrapAngle – нормализует угол, сопоставляя его с диапазоном от 0 до 360 градусов для корректного использования в расчетах.

И теперь можно перейти к двум функциям, которые вычисляют и применяют силы, необходимые для управления БПЛА, и используют ПИД-регулятор для стабилизации наклона и поворота, это функции AddMotorForce и AddForceToPropeller

Как ранее уже было написано, в данной программе для стабилизации БПЛА используется ПИД-регулятор, написанный в отдельном файле.

ПИД-контроллер, или пропорционально-интегрально-дифференциальный контроллер, является важной частью управления в динамических системах. Он используется для поддержания целевого состояния системы за счет непрерывного расчета ошибки как разницы между заданным и фактическим состояниями. В контексте симулятора для БПЛА ПИД-контроллер играет ключевую роль в стабилизации полета, обеспечивая плавность и точность управления.

Пропорциональная составляющая отвечает за реакцию на текущую ошибку. Чем выше коэффициент, тем более резкой будет коррекция, что может привести к нестабильности, если значение слишком велико.

Интегральная составляющая заботится о накоплении прошлых ошибок. Она позволяет уменьшить систематические отклонения, усиливая коррекцию, если ошибка остается, но без чрезмерной резкости, что делает управление более стабильным.

Дифференциальная составляющая реагирует на скорость изменения ошибки, помогая смягчать переходные процессы и предотвращать «переход» через целевой параметр.

Настройка этих параметров является критически важной задачей для достижения максимальной точной имитации полета реального БПЛА.

Теперь в данном симуляторе можно осуществлять управление БПЛА, но при его перемещении камера остается зафиксированной на одном месте. Чтобы реализовать вид от первого лица, как изначально и планировалось, необходимо создать ещё один скрипт, который будет применён к камере.

Данный код фиксирует камеру в Unity на определенной части БПЛА и поворачивает её в ту сторону, в которую он направлен. Также тут применяется стабилизации для уменьшения тряски камеры.



Рис. 3. Демонстрация работы симулятора

Теперь остается только создать окружение, в котором данный БПЛА будет летать, и можно считать, что симулятор готов. Но поскольку создание окружения сводится к простой установке готовых моделей, в данной статье этот процесс рассматриваться не будет.

После создания окружения, в котором будет летать БПЛА, и запуска проекта симулятор будет иметь вид, представленный на рисунке 3.

Разработанный симулятор для обучения операторов БПЛА в среде Unity демонстрирует потенциал для эффективной подготовки специалистов. Использование Input System для интеграции контроллера Radiomaster TX12 обеспечивает реалистичное управление, позволяя операторам освоить навыки управления дроном в безопасной виртуальной среде. Применение ПИД-регуляторов для стабилизации БПЛА в симуляторе способствует более точному моделированию физики полета, приближая виртуальный опыт к реальным условиям. Возможность настройки входных сигналов контроллера, включая нормализацию и инверсию, обеспечивает гибкость и адаптивность симулятора к различным типам оборудования и индивидуальным предпочтениям операторов. Создание скрипта для камеры с фиксацией на БПЛА и стабилизацией тряски повышает иммерсивность обучения и позволяет операторам отрабатывать навыки визуального наблюдения. В целом, симулятор предоставляет платформу для обучения и тренировки операторов БПЛА, способствуя снижению рисков и повышению безопасности в реальных условиях эксплуатации.

Заключение

Заключение данной работы подводит итоги разработки симулятора для обучения операторов беспилотных летательных аппаратов (БПЛА) с использованием платформы Unity и интеграции современных технологий управления. Созданный симулятор представляет собой эффективный и инновационный инструмент, обеспечивающий высококачественную подготовку операторов, на что указывают возможности адаптации сценариев, отработка действий

в экстремальных ситуациях и высокий уровень визуализации.

Симулятор демонстрирует возможности применения PID-контроллера для стабилизации полета, улучшая маневренность и отзывчивость БПЛА в виртуальной среде. Корректная настройка его параметров – пропорциональной, интегральной и дифференциальной составляющих – является критическим аспектом для достижения реальной имитации полета. Это позволяет не только подготовить новых операторов, но и повысить квалификацию опытных специалистов.

Перспективы дальнейшей работы включают внедрение новых функциональных возможностей на основе отзывов пользователей, что позволит адаптировать симулятор под разнообразные задачи и требования операционных условий. Таким образом, разработанный симулятор не только минимизирует риски, связанные с реальными полетами, но и способствует формированию высококвалифицированных операторов, готовых к выполнению сложных задач в условиях быстрого технологического прогресса.

Список литературы

1. Будкин Е.Г., Постников А.А., Лагутина Е.И. Роль симулятора в подготовке оператора БПЛА: вызов современности процессу обучения в военном вузе // Вестник военного образования. 2023. № 4. С. 27-30.
2. Ким В.А. Роль симуляторов в обучении и оценке квалификационных требований пилотов // Universum. 2024. 8(125). URL: <https://universum.com/ru/tech/archive/item/18073> (дата обращения: 24.06.2024).
3. Исаев Р.Р., Исмагилов А.А. Методические аспекты обучения пилотированию БПЛА с применением компьютерных симуляторов // Вестник Башкирского государственного педагогического университета им. М. Акмуллы. 2019. № 2 (50). С. 6-11.
4. Дудырев Ф.Ф., Максименкова О.В. Симуляторы и тренажеры в профессиональном образовании: педагогические и технологические аспекты // Вопросы образования. 2020. № 3. С. 255-276.
5. Unity Technologies. Документация Unity. [Электронный ресурс]. Режим доступа: <https://docs.unity3d.com/Manual/index.html> (дата обращения 17.09.2024).
6. Ezhil V.R., Sriram B.S.R., Vijay R.C., Yeshwant S., Sabareesh R.K., Dakkshesh G., Raffik R. Investigation on PID controller usage on Unmanned Aerial Vehicle for stability control // Materials Today: Proceedings. 2022. Vol. 66, Is. 3. P. 1313–1318.
7. Petukhov I.V., Kurasov P.A., Steshina L.A., Tanryverdiev I.O. Analysis of the Effectiveness of Moving Object Management in Virtual Reality // 11th International Conference on Systems and Control (ICSC). Sousse, Tunisia, 2023. P. 862–865. DOI: 10.1109/ICSC58660.2023.10449842.