



Проектирование квадрокоптера на базе интегрированной модельной среды

М. В. Чугунов*, **И. Н. Полунина**, **М. А. Попков**
ФГБОУ ВО «МГУ им. Н. П. Огарёва» (г. Саранск, Россия)
**m.v.chugunov@mail.ru*

Введение. В статье представлен междисциплинарный подход к проектированию летающего робота (квадрокоптера), основанный на использовании интегрированной модельной среды. Процесс проектирования реализуется как создание моделей разного типа: натурной и виртуальной.

Материалы и методы. Под виртуальной моделью понимается совокупность математических, алгоритмических, программных и 3D-моделей, обеспечивающих ее функционирование в виртуальной среде. Проектное решение представляет собой комплект конструкторско-технологической документации, включающей в себя интегрированную модель проектируемого объекта, компоненты которой связаны друг с другом. Натурная часть интегрированной модельной среды включает в себя следующие компоненты: несущую силовую систему; корпусные детали; электромеханические и электронные системы управления, навигации, телеметрии и сенсорики. Для несущих систем и корпусных деталей используются полиамидные бionicеские конструкции, приобретенные и напечатанные на 3D-принтере; базовым элементом электронной системы является полетный контроллер ArduPilot (ArduCopter). Виртуальная среда моделирования формируется на базе CAD/CAE/CAM/PDM/PLM SolidWorks (Motion, Simulation). Основными инструментами, используемыми для создания связей между моделями разного типа и уровня, являются СОМ-технология, API CAD/CAE/CAM/PDM/PLM-системы, MS Visual Studio C++, позволяющие разработать единый интерфейс для управления полетом и планирования траектории как в реальной, так и в виртуальной среде.

Результаты исследования. Разработана интегрированная (натурная и виртуальная) модельная среда для квадрокоптера. На данной основе получено проектное решение в виде реального объекта и его виртуальной модели. Состояние и поведение этих объектов контролируется и управляется программным обеспечением, имеющим доступ как к реальному объекту, так и к его 3D-модели. Полученный результат можно рассматривать как инструмент инжиниринга для решения широкого спектра научно-технических и производственных задач: проведения дефектоскопии, диагностики аварийных ситуаций, 3D-сканирования удаленных и труднодоступных объектов.

Обсуждение и заключение. В статье показана эффективность подхода к проектированию как к процессу создания междисциплинарных моделей разного типа и уровня. При этом особую значимость приобретает проблема интеграции этих моделей в единое целое за счет формирования двунаправленных ассоциативных связей. Дальнейшего развития требуют технологические (программные) средства синхронизации состояния и поведения натурной и виртуальной моделей объекта проектирования.

Ключевые слова: квадрокоптер, интегрированная модельная среда, виртуальная модель, натурная модель, двунаправленная ассоциативная связь, система автоматизированного проектирования, СОМ-технология, API-программирование



Для цитирования: Чугунов М. В., Полунина И. Н., Попков М. А. Проектирование квадрокоптера на базе интегрированной модельной среды // Инженерные технологии и системы. 2019. Т. 29, № 2. С. 169–186. DOI: <https://doi.org/10.15507/2658-4123.029.201902.169-186>

The Quadcopter Design Based on Integrated Model Environment

M. V. Chugunov*, I. N. Polunina, M. A. Popkov

National Research Mordovia State University (Saransk, Russia)

**m.v.chugunov@mail.ru*

Introduction. The deals with the multi/interdisciplinary approach to designing the unmanned aerial vehicle (quadcopter) based on the use of the integrated model environment. The designing process is implemented as creating different types of models: natural (physics) and virtual.

Materials and Methods. The virtual model is understood to be a set of mathematical, algorithmic, program and 3D models maintaining its functioning in virtual environment. The design decision represents a set of the design-technology documentation including the integrated model of the designed project, whose components are connected with each other. The natural (physics) part of the integrated model environment includes the following components: a carrier system, shell details, electromechanical and electronic systems for controlling navigation, telemetry and sensory. For the carrier systems and shells there used polyamide bionic parts, which are purchased and printed on the 3D printer; the basic element of electronic system is the flight controller ArduPilot (ArduCopter). The virtual environment of modelling is formed on the basis of CAD/CAE/CAM/PDM/PLM SolidWorks (Motion, Simulation). The main tools, used for creating the communications between models of different types and levels, are the COM technology, API CAD/CAE/CAM/PDM/PLM system, MS Visual Studio C++, which allow developing the integrated interface for controlling the flight and planning the trajectory in the real and virtual environment.

Results. The integrated (natural and virtual) model environment for the quadcopter is developed. On this basis, the design decision in the form of a real object and its virtual model is made. The state and behaviour of these objects is controlled and guided by the software having access both to a real object and to its 3D model. The received result can be considered as the tool of engineering for the solution of a wide range of scientific, technical and production tasks: performing defectoscopy, diagnosing emergencies, and 3D-scanning remote and hard-to-reach objects.

Discussion and Conclusion. The research shows the efficiency of the approach to designing as to process of creating the multi/interdisciplinary models of different types and levels. At the same time, the problem of integrating these models into a coherent whole by forming bidirectional associative communications has assumed particular importance. The technological (program) means for synchronizing a state and behaviour of the natural and virtual models of design objects require further development.

Keywords: quadcopter, integrated model environment, virtual model, full-scale model, bidirectional associative relation, computer-aided engineering system, COM technology, API programming

For citation: Chugunov M.V., Polunina I.N., Popkov M.A. The Quadcopter Design Based on Integrated Model Environment. *Inzhenernyye tekhnologii i sistemy* = Engineering Technologies and Systems. 2019; 29(2):169-186. DOI: <https://doi.org/10.15507/2658-4123.029.201902.169-186>

Введение

Проблема разработки интегрированных модельных сред становится все более актуальной, а популярность и эффективность их использования в различных научно-технических, естественнонаучных и производственных областях стремительно возрастает.

Структуру, функционал и свойства интегрированной модели наглядно иллюстрирует разработка компании Dassault Systèmes для робота-гуманоида (рис. 1)¹.

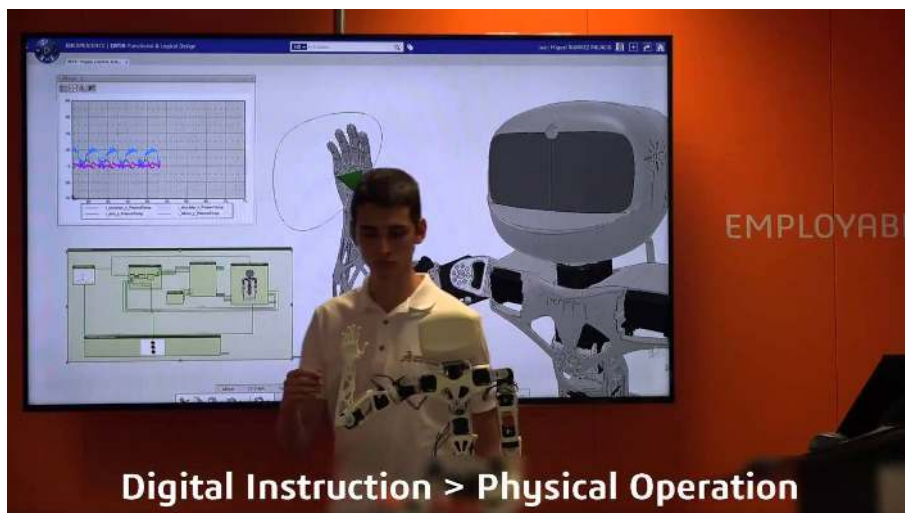
По своей структуре интегрированная модель включает в себя три части: натурную, виртуальную и систему управления обеими моделями, которая поддерживает также двунаправленные ассоциативные связи между моделями разного типа. Реализованный в указанном случае функционал (рис. 1) состоит в имитации двигательных функций человека одновременно в двух средах: реальной (натурной) и виртуальной. Основными свойствами данной интег-

рированной модели являются синхронизированность и ассоциативность состояния и поведения моделей разного типа. Под ассоциативностью мы понимаем изменение (в том числе и по структуре) одной модели, обусловленное изменением состояния другой модели.

В данной статье мы рассматриваем квадрокоптер не как объект и инструмент практического применения, область которого в последнее время стремительно расширяется², а как объект исследования, моделирования и проектирования.

В целом следует указать несколько научных школ, уделяющих серьезное внимание проектированию и исследованию квадрокоптеров во всех возможных аспектах:

– лаборатория профессора Виджая Кумара³, университет Пенсильвании, США (Department of Mechanical Engineering and Applied Mechanics, School of Engineering and Applied Science, University of Pennsylvania);



Р и с. 1. Интегрированная модель

F i g. 1. The integrated model

¹ Juan Miguel – Dassault Systèmes Trainee – 2015 – YouTube. URL: <https://www.youtube.com/watch?v=ESWaCrw8rg>; Learning Virtual Universes with Poppy Humanoid – YouTube. URL: <https://www.youtube.com/watch?v=mMmH2Dhh-94>

² 433056.pdf. URL: <http://preview.thenewsmarket.com/Previews/PWC/DocumentAssets/433056.pdf>

³ Vijay Kumar Lab. URL: <https://www.kumarrobotics.org>

– аэрокосмический институт университета Торонто, профессор Анжела Шуллиг⁴, Канада (University of Toronto, Institute for Aerospace Studies);

– Московский авиационный институт (национальный исследовательский университет) (MAI Drone school⁵);

– Московский государственный технический университет им. Н. Э. Баумана⁶.

Целью данной статьи является разработка модельной среды для проектирования квадрокоптера, которая обеспечивала бы разработку интегрированной модели, по своей структуре, функционалу и свойствам аналогичную указанную (рис. 1, [1–3]).

Важной особенностью рассматриваемого подхода является то, что проектирование и управление объектом реализуется в одной и той же CAD/CAE/CAM/PDM/PLM-системе, используемой инженером в своей повседневной деятельности. На данный момент авторам статьи не известно о решении задачи построения интегрированных моделей для квадрокоптера в такой постановке.

Обзор литературы

Интегрированные модели уже давно находятся в сфере внимания специалистов разного профиля; в частности, теоретические и практические аспекты этой проблемы концептуально рассмотрены в работе И. М. Макарова, В. З. Рахманкулова и А. А. Ахрема [1]. Что касается проектирования технических систем, то прогресс, достигнутый в области междисциплинарного моделирования, де-факто привел к новому пониманию процесса проектирования, который рассматривается как созда-

ние взаимосвязанных моделей разного типа. Данный подход дает ряд преимуществ: возможность оперативной оценки эффективности проектных решений и адекватности результатов как натурального, так и численного экспериментов, а также возможность параметризации и масштабируемости, мультиагентности, многотерминальности и т. д. [2; 3].

В данном контексте летающие роботы (квадрокоптеры) привлекают внимание исследователей разного профиля в междисциплинарной и междисциплинарной постановке. Математический аспект задачи включает в себя: аэродинамическую [4; 5], кинематическую и динамическую [5–7] модели для твердого тела с шестью степенями свободы, а также модели стабилизации и траекторного управления [Там же.]. Как правило, определяются и учитываются лишь аэродинамические параметры пропеллеров, но не корпусных и несущих систем конструкции, а уравнения динамики квадрокоптера линеаризуются за счет предположения о малости угловых координат и взаимных производений угловых скоростей [7; 8]. В качестве алгоритмов автоматического управления используются пропорционально-дифференциальные (ПД) [6; 8] и пропорционально-дифференциально-интегральные (ПИД) регуляторы [7; 8].

В своей конструкторской части рассматриваемая задача отличается широким спектром известных решений, выполненных на базе открытых платформ. Подробный сравнительный анализ восьми широко известных платформ приведен в работе И. А. Калинова⁷.

⁴ Welcome to the Dynamic Systems Lab. Dynamic Systems Lab. Prof. Angela Schoellig. URL: <http://www.dynsyslab.org/prof-angela-schoellig>

⁵ Беспилотный комплекс Luftera LQ-4. URL: https://mai.ru/science/dev/index.php?ELEMENT_ID=80837; Комплекс мини-БЛА четырехосной вертолетной схемы. URL: https://mai.ru/science/dev/index.php?ELEMENT_ID=9955; Микробеспилотный летательный аппарат. URL: https://mai.ru/science/dev/index.php?ELEMENT_ID=9901

⁶ Научно-учебный комплекс Фундаментальные науки МГТУ им. Н. Э. Баумана – Научные лаборатории. URL: <http://www.fn.bmstu.ru/research-section-sec-fs/research-fs>

⁷ Калинов И. А. Особенности применения платформ конструирования квадрокоптеров с доступным исходным кодом для решения задач мониторинга открытых пространств при помощи построения интеллектуальных систем // Новые информационные технологии в автоматизированных системах : мат-лы XIX науч.-практ. семинара. М. : МИЭМ НИУ ВШЭ, 2016. С. 245–251.

В других статьях [9; 10] проектирование в конструкторском, технологическом и функциональном аспекте выполнено для квадрокоптера, управляемого с пульта с возможностью стабилизации полета.

Исследование систем управления квадрокоптерами, основанное на сочетании математических методов с натурным экспериментом, выполнено А. А. Гоголевым [11].

В качестве инструментов численного анализа математической модели квадрокоптера в указанных работах используются системы MathCAD, Matlab, Simulink и Universal Mechanism. CAD/CAE (в частности, SolidWorks) используется учеными [12] в качестве среды проектирования и платформы для разработки специализированного программного обеспечения с целью решения задач рассматриваемого класса.

Материалы и методы

Конструирование несущей системы и сборка аппаратной части

Проектирование несущей системы квадрокоптера осуществлялось в среде SolidWorks. За основу взята конструкция F-450 на базе APM. Было рассмотрено несколько проектных решений, которые подробно изложены авторами в другой статье [13]. На рис. 2 показано фотореалистичное изображение 3D-модели конструкции⁸.

Математическая модель и программная реализация

Положение квадрокоптера в пространстве определялось координатами x , y , z центра масс аппарата в неподвижной системе декартовых координат и тремя углами поворота вокруг главных центральных осей инерции квадрокоптера в подвижной системе ко-



Р и с. 2. Фотореалистичное изображение 3D-модели сборки
F i g. 2. The photoview image for 3D assembly

⁸ Сборка.avi – Google Диск. URL: https://drive.google.com/file/d/1EXr8hMO3_vpRoHHXdcY9tvS-CK8sVwB6/view?usp=sharing

ординат (xx , yy , zz): ψ – угол рыскания; φ – угол крена; θ – угол тангажа (рис. 3).

Уравнения динамики, описывающие движение аппарата, таковы [4–8]:

$$\begin{aligned} \ddot{z} &= -g + (\cos\theta\cos\varphi)\frac{u_1}{m}, \\ \ddot{y} &= (-\cos\psi\sin\varphi + \sin\psi\cos\varphi\sin\theta)\frac{u_1}{m}, \\ \ddot{x} &= (\sin\varphi\sin\psi + \cos\psi\cos\varphi\sin\theta)\frac{u_1}{m}, \\ \ddot{\varphi} &= \frac{u_2}{J_{xx}} - \frac{J_{zz}-J_{yy}}{J_{xx}}\dot{\theta}\dot{\psi}, \\ \ddot{\theta} &= \frac{u_3}{J_{yy}} - \frac{J_{xx}-J_{zz}}{J_{yy}}\dot{\varphi}\dot{\psi}, \\ \ddot{\psi} &= \frac{u_4}{J_{zz}}. \end{aligned}$$

Здесь \ddot{z} – линейное ускорение; $\ddot{\varphi}$, $\ddot{\theta}$, $\ddot{\psi}$ – угловые ускорения; g – ускорение свободного падения; m – масса аппарата; J_{xx} , J_{yy} , J_{zz} – главные моменты инерции аппарата; u_1 , u_2 , u_3 , u_4 – управляющие параметры: u_1 – сила, действующая вдоль оси zz аппарата; u_2 , u_3 , u_4 – моменты относительно осей xx , yy , zz . Через

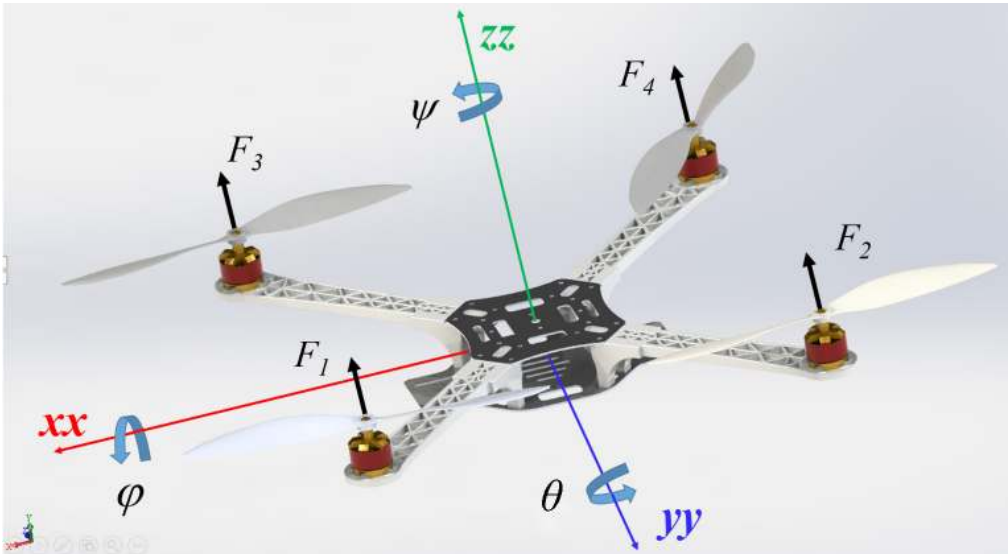
силы тяги F_1 , F_2 , F_3 , F_4 , создаваемые винтами квадрокоптера, управляющие параметры определяются следующим образом:

$$\begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -l & 1 & l \\ -l & l & -l & l \\ -\lambda & \lambda & -\lambda & \lambda \end{bmatrix} \begin{bmatrix} F_1 \\ F_2 \\ F_3 \\ F_4 \end{bmatrix},$$

где l – расстояние от осей винтов до центра тяжести квадрокоптера; λ – коэффициент пропорциональности между тягами винтов и реактивными моментами вращения относительно осей моторов.

Для планирования полета чаще всего используются вариационные методы, основанные на минимизации функционала, который в общем случае записывается в следующем виде [14]:

$$L = \int_{t_0}^t (\mu_r \frac{d^{(k)}r^2}{dt} + \mu_\psi \dot{\psi}^2) dt, \quad (2)$$



Р и с. 3. Системы координат квадрокоптера

F i g. 3. The quadcopter coordinate systems

где $r(t) = (x(t) \ y(t) \ z(t))^T$; μ_x и μ_y – коэффициенты, введенные для приведения подынтегрального выражения к безразмерному виду; $k = 2, 4$ – порядки учитываемой производной.

Рассмотрим случай $L = \int_0^T (\ddot{z})^2 dt$, т. е. найдем траекторию для вертикального полета из условия минимума ускорения (minimum acceleration). На рис. 4 представлены результаты моделирования этой задачи в среде SolidWorks Motion с соответствующими граничными условиями для z и \dot{z} в начальной ($t = t_0$) и конечной ($t = T$) точках полета.

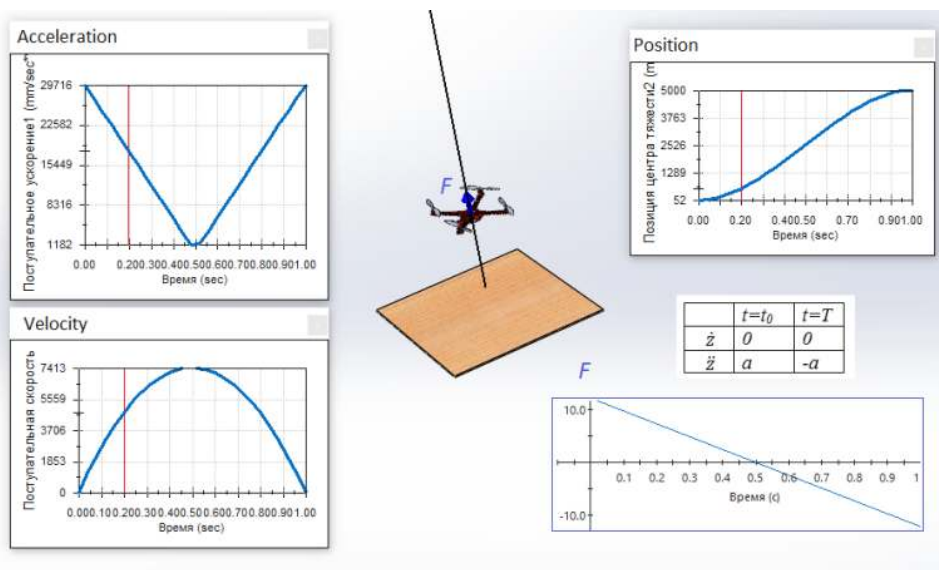
Решение следует из соответствующего уравнения Эйлера – Лагранжа в виде полинома $z(t) = -\frac{a}{3T}t^3 + \frac{a}{2}t^2$, где a – модуль ускорения в начальной и конечной точке; T – время полета. На рис. 4 показаны траектория движения,

а также графики скорости, ускорения и силы тяги (F)⁹.

Аналогичное решение получено для функционала $L = \int_0^T (\ddot{z})^2 dt$ (условие minimum jerk) в виде полинома $z(t) = 30t^5 - 75t^4 + 50t^3$. Граничные условия и результаты¹⁰ показаны на рис. 5.

В работе А. А. Кочкарова и Р. Т. Агишева [15] приведены результаты подробного сравнительного анализа эффективности методов управления, реализованных для кривых (полиномов), полученных на основе минимизации функционала при разных значениях k , т. е. при $k = 2$ (minimum acceleration), $k = 3$ (minimum jerk), $k = 4$ (minimum snap).

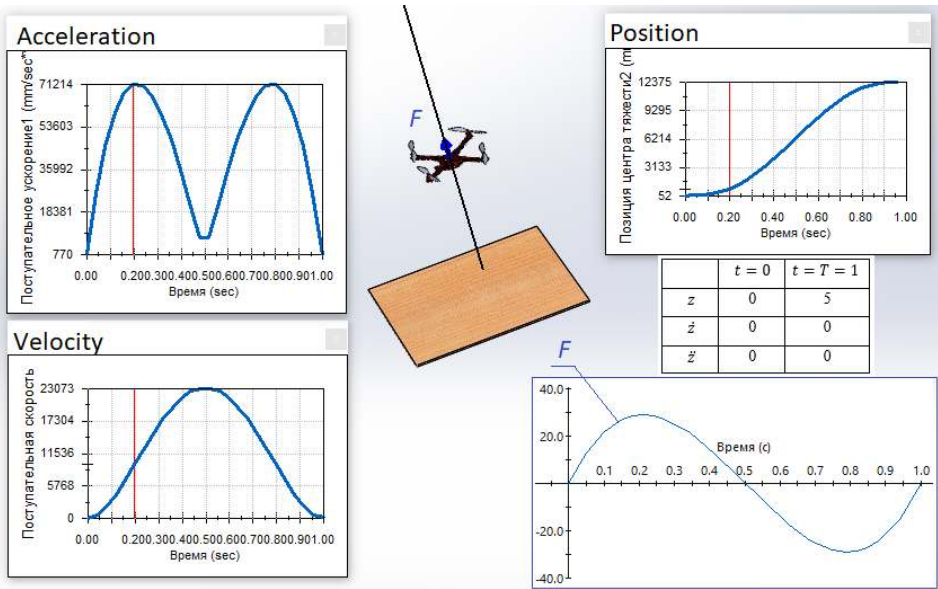
Отмечено, что кривые (сплайны) более высоких порядков гарантируют прохождение узловых точек с более высокой степенью точности, но сплайны



Р и с. 4. Результаты моделирования вертикального взлета из условия minimum acceleration
F i g. 4. The modelling results for vertical takeoff with minimum acceleration condition

⁹ Acc1_z_1c.avi – Google Диск. URL: <https://drive.google.com/file/d/1WthMumBAmIDkc2QoDk1OXKOxephi9P-n/view?usp=sharing>

¹⁰ Jerk_1c.avi – Google Диск. URL: <https://drive.google.com/file/d/1msFSlb-DgpXwTOWnI3XCyfHzj3ujfARo/view?usp=sharing>



Р и с. 5. Результаты моделирования вертикального взлета из условия minimum jerk
 F i g. 5. The modelling results for vertical takeoff with minimum jerk condition

более низкого порядка дают меньшую суммарную ошибку по всей траектории.

Рассмотрим решение задачи планирования полета в общем случае из условия minimum jerk. Здесь для каждого координатного направления необходимо определить шесть констант, соответствующих заданным граничным условиям.

Для решения задачи планирования траектории удобно использовать сплайны. В среде SolidWorks в качестве сплайнов используются NURBS-кривые [16], которые в частном случае четырех узловых точек представляют собой кривые Безье.

В практическом смысле следует указать два отличия кривых Безье от NURBS:

- 1) изменение положения одной ключевой точки приводит к перестраиванию всей кривой;
- 2) кривые Безье значительно проще, чем NURBS, задаются аналитически.

Оба эти свойства определяют наш выбор для решения задачи планиро-

вания полетом в пользу кривых Безье. Аналитически кривые задаются в параметрической форме [16]:

$$\begin{aligned}
 x(t) &= \sum_{i=0}^N \Phi_{N_i}(t) x_i, \\
 y(t) &= \sum_{i=0}^N \Phi_{N_i}(t) y_i, \\
 z(t) &= \sum_{i=0}^N \Phi_{N_i}(t) z_i, \quad (3)
 \end{aligned}$$

где $\Phi_{N_i}(t) = C_N^i t^i (1-t)^{N-i}$; здесь $C_N^i = \frac{N!}{i!(N-i)!}$; $0 \leq t \leq 1$. Здесь $i = 0, N-1$; $N = n-1$; n – количество узловых точек.

Основное преимущество, которое позволяет получить использование кривых Безье, – простота формирования граничных условий для полета. Так, в случае кривой пятого порядка необходимо задать шесть ключевых точек ($n = 6$), которые определяют сплайн

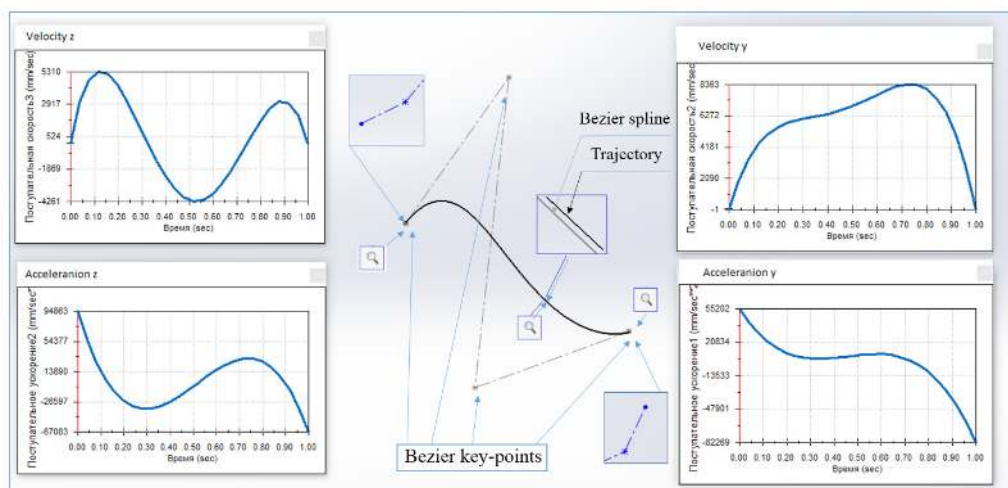
геометрически. Более того, взаимное расположение точек позволяет задать граничные условия, аналогичные задаче (рис. 5). В частности, при малом удалении второй ($i = 1$) от первой ($i = 0$), а пятой ($i = 4$) от шестой, последней по порядку точки ($i = 5$), задается малость скоростей полета в этих точках для каждого координатного направления.

На рис. 6 показаны результаты планирования траектории полета по кривой Безье с шестью ключевыми точками в вертикальной плоскости. Показаны также три области многократного зуммирования, помеченные символом «лупа»: для областей в окрестности начальной и конечной точек сплайна. Видно, что пары первой и второй, а также последней и предпоследней точек близко расположены друг к другу. Третья область показывает исходную кривую (Bezier Spline) и фактическую траекторию полета (Trajectory), построенную в результате анализа движения, выполненного в среде SolidWorks Motion; эти кривые близки друг к другу. В качестве

внешних воздействий рассматриваются управления, построенные на основе плоской системы управления [17], а функции $y(t)$ и $z(t)$ заданы уравнениями (3). На рис. 6 показана траектория движения, графики скоростей и ускорений центра тяжести аппарата в каждом из координатных направлений¹¹.

В практическом плане интерес представляет решение задачи управления и стабилизации полета по заданной траектории по принципу ПД- (ПИД)-регулирования [18]. В этом случае целью управления является асимптотическое снижение (до нуля) отклонения аппарата от заданной траектории. В работе авторов [19] приведен ряд примеров решения простейших задач данного типа в среде SolidWorks Motion.

Наиболее распространенным и популярным на сегодняшний день программным обеспечением, используемым для настройки и управления летательным аппаратом, оснащенным полетным контроллером АРМ, является Mission Planner (Mission Planner Ground



Р и с. 6. Результаты моделирования полета из условия minimum jerk в вертикальной плоскости
F i g. 6. The modelling results for vertical fly with minimum jerk condition in the vertical plane

¹¹ Cube.avi – Google Диск. URL: <https://drive.google.com/file/d/1FXAMI1bVXfUSohTGiwE5RR-aGAX5BkHi/view?usp=sharing>

Control Station¹²). Проект является открытым и реализован на принципах GIT (Global Information Tracker), распределенной системы контроля версий, предназначенной для коллективной разработки¹³.

Так же, как и в Mission Planner, мы будем использовать метод ПД-регулирования. Рассмотрим решение задачи в общем виде. Потребуем удовлетворения координат центра тяжести квадрокоптера следующему уравнению:

$$(\dot{r}_d(t) - \dot{r}_c(t)) + k_d e_v + k_p e_p = 0,$$

где

$$\begin{aligned} e_p &= r_d(t) - r(t), \\ \dot{e}_p &= \dot{r}_d(t) - \dot{r}(t), \\ r(t) &= (x(t) y(t) z(t) \psi(t))^T, \end{aligned} \quad (4)$$

где $r_d(t)$ – желаемый радиус-вектор для центра тяжести, дополненный углом рыскания; $r(t)$ – фактический радиус-вектор; $\ddot{r}_c(t)$ – задаваемое ускорение, вычисленное управляющим контроллером; k_d и k_p – матрицы коэффициентов ПД-регулятора.

Полный набор линеаризованных соотношений для задачи управления по траектории приобретает следующий вид [15; 18]:

$$u_1 = m(g + \dot{z}_d + k_{vz}(\dot{z}_d - \dot{z}) + k_{pz}(z_d - z)), \quad (5)$$

$$\dot{x}_c = \dot{x}_d + k_{px}(x_d - x) + k_{vx}(\dot{x}_d - \dot{x}),$$

$$\dot{y}_c = \dot{y}_d + k_{py}(y_d - y) + k_{vy}(\dot{y}_d - \dot{y}),$$

$$\varphi_c = \frac{1}{g}(\dot{x}_c \sin(\psi_d) - \dot{y}_c \sin(\psi_d)),$$

$$\theta_c = \frac{1}{g}(\dot{x}_c \cos(\psi_d) + \dot{y}_c \sin(\psi_d)),$$

$$u_2 = k_{v\varphi}(\dot{\varphi}_d - \dot{\varphi}) + k_{p\varphi}(\varphi_d - \varphi), \quad (6)$$

$$u_3 = k_{v\theta}(\dot{\theta}_d - \dot{\theta}) + k_{p\theta}(\theta_d - \theta), \quad (7)$$

$$u_4 = k_{v\psi}(\dot{\psi}_d - \dot{\psi}) + k_{p\psi}(\psi_d - \psi). \quad (8)$$

Здесь u_1 – необходимая сила тяги; u_2 , u_3 , u_4 – необходимые моменты относительно главных осей аппарата.

Результаты исследования

Математическая модель (5–8) была реализована как для натурной, так и для 3D-модели в среде программирования MSVisual Studio C++ (2015)¹⁴ в виде приложения Windows, названного нами swMotion.

На рис. 7, а представлен пользовательский интерфейс приложения для решения задачи планирования траектории в среде, интегрированной с SolidWorks; на рис. 7, б – блок-схема алгоритма решения этой задачи:

– *COM-initialization* – инициализация COM-интерфейса;

– *Feature Cycle* – цикл по фитчерам 3D-модели полетной среды, где сплайн, задающий траекторию, имеет характерное название;

– *Get Data: spline key-points* – импорт параметров сплайна;

– *Preparing moving parameters and drawing knots* – расчет полетных режимов; графические процедуры для узлов и промежуточных точек сплайна; перевод координат из декартовых в географические¹⁵; экспорт данных в файл формата WayPoints MissionPlanner¹⁶, который загружает созданную на основе этого файла программу управления полетом, исполняемую контроллером натурной модели.

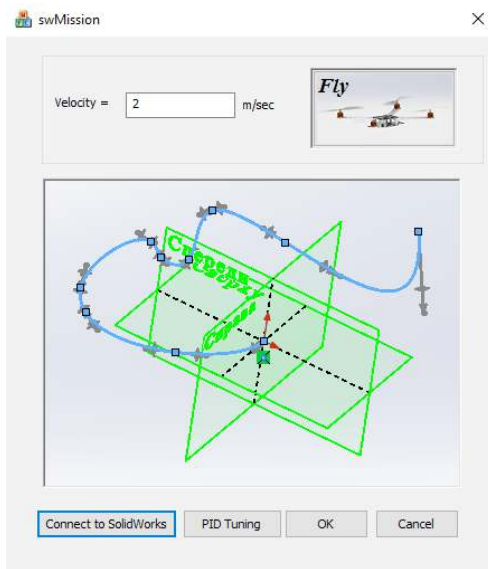
¹² Mission Planner Home – Mission Planner documentation. URL: <http://ardupilot.org/planner>

¹³ Building Mission Planner with Visual Studio – Dev documentation. URL: <http://ardupilot.org/dev/docs/buildin-mission-planner.html>

¹⁴ Download Распространяемый пакет Visual C++ для Visual Studio 2015 from Official Microsoft Download Center. URL: <https://www.microsoft.com/ru-ru/download/details.aspx?id=48145>

¹⁵ u-blox. URL: <https://www.u-blox.com/en>

¹⁶ Search – Dev documentation. URL: http://ardupilot.org/dev/search.html?q=Way+Points+&check_keywords=yes&area=default#

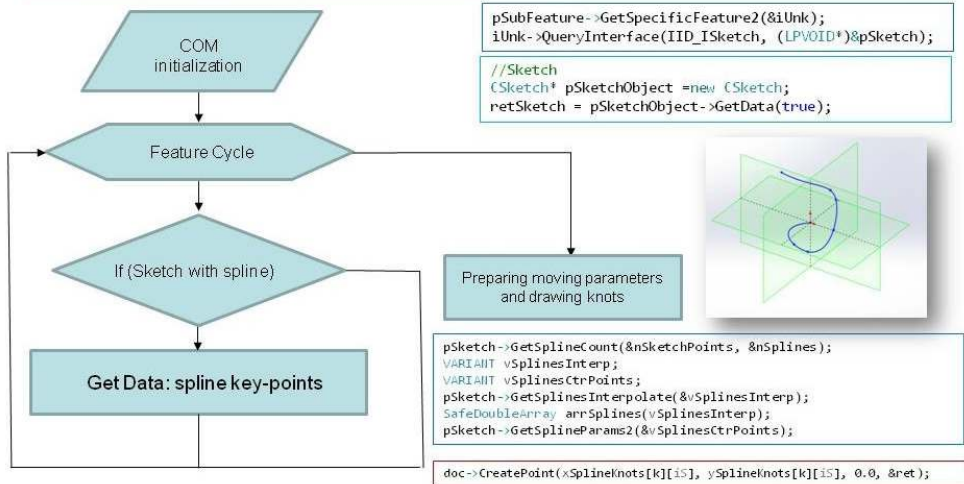


a)

#import "C:\Program Files\SolidWorks Corp\SolidWorks\slidworks.tlb" raw_interfaces_only, raw_native_types, no_namespace, named_guids

```
//Initialize COM
CoInitialize(NULL);
//Create an instance of SolidWorks
doc=NULL;
swApp.CoCreateInstance(L"SolidWorks.Application", NULL, CLSCTX_LOCAL_SERVER);
swApp->get_IActiveDoc2(&doc);
swApp->put_Visible(VARIANT_TRUE);
```

```
//Use ATL smart pointers
CComPtr<ISolidWorks> swApp;
CComPtr<IModelDoc2> doc=NULL;
CComPtr<IFeature> pFeature=NULL;
CComPtr<IFeature> pSubFeature=NULL;
CComPtr<ISketchManager> pSketchMgr=NULL;
```



b)

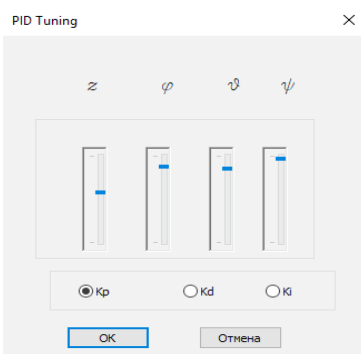
Р и с. 7. Приложение swMotion: а) интерфейс пользователя; б) блок-схема алгоритма
 F i g. 7. swMotion application: a) the user interface; b) algorithm flowchart

Диалоговая панель (рис. 8, а) предназначена для настройки коэффициентов ПИД-регулирования K_p , K_d и K_i для пропорционального, дифференциального и интегрального регулирования соответственно (коэффициенты, соответствующие положению радиокнопки K_i , в настоящее время зарезервированы, но не реализованы).

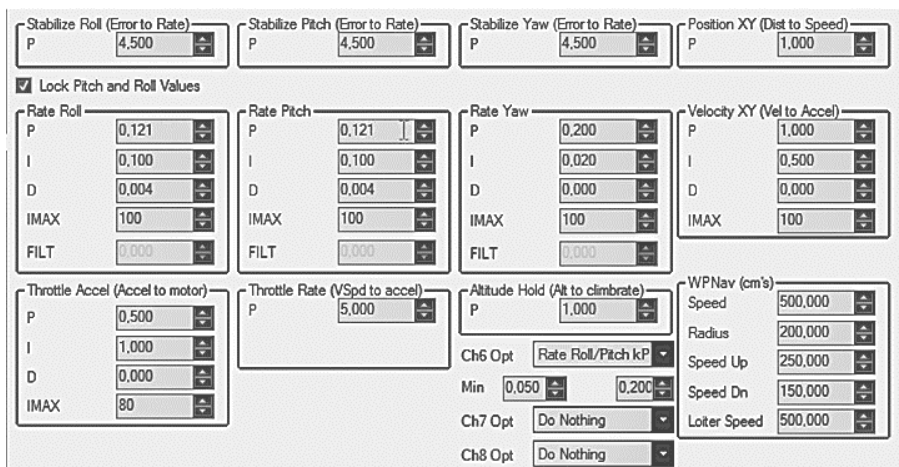
В качестве значений коэффициентов по умолчанию принимаются значения, соответствующие настройкам натурной модели (MissionPlanner; рис 8, б), а регуляторы запрограммированы на возможное изменение этих значений на 50 % вверх и вниз.

На рис. 9 показан результат решения задачи планирования и управления полетом по заданной траектории в виде сплайна Безье (показана кривая и ее управляющий многоугольник); сплошная кривая – действительная траектория¹⁷.

Среди особенностей программной реализации рассматриваемого подхода следует указать следующие. Алгоритм, реализующий математическую модель (5–8), требует многократной коррекции управлений u_1 , u_2 , u_3 и u_4 по времени, а также многократного обращения к результатам расчетов для параметров движения.



а)

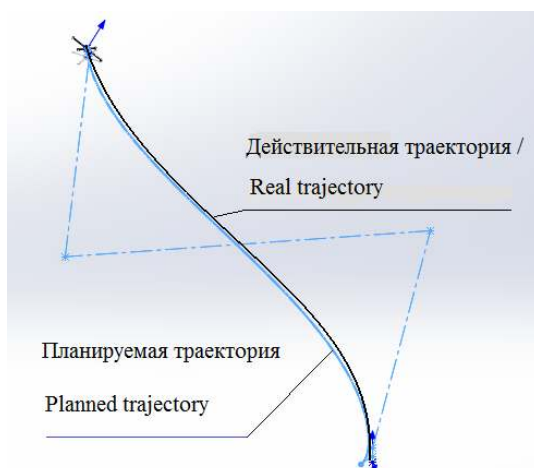


б)

Р и с. 8. Настройка ПИД для виртуальной (а) и натурной (б) модели

Fig. 8. PID tuning for virtual (a) and physical (b) model

¹⁷copter_Result.avi – Google Диск. URL: https://drive.google.com/file/d/1zpGK8rY1jJNDX7HR6SlvSIM45K_rHu/view?usp=sharing



Р и с. 9. Результаты планирования и управления полетом
F i g. 9. The results for planning and controlling the flight

На рис. 10, а приведен фрагмент кода C++ COM для решения первой задачи на примере управления u_1 : массив `spData_u1` содержит как дискретные значения времени `Time[idx]`, так и соответствующие им значения управления `u1[idx]`. Достоинством данного подхода является то, что определенные таким образом значения управления при экспорте в SolidWorks Motion сглаживаются сплайном.

Для решения второй задачи, т. е. для импорта в приложение результатов расчетов, авторы попытались использовать инструмент SolidWorks «Датчик». Дан-

ный инструмент успешно использовался ранее в среде SolidWorks Simulation [20]. Однако выяснилось, что в API SolidWorks Motion отсутствуют адекватные средства для импорта результатов расчетов с датчиков. Задачу удалось решить с использованием функций непосредственного доступа к результатам расчетов. На рис. 10, б показан фрагмент кода для определения текущих координат центра тяжести аппарата (CM Position).

Обсуждение и заключение

На рис. 11 показаны два основных метода создания интегрированной мо-

```

pSubFeature->GetDefinition(&pDisp);
pDisp->QueryInterface(&pFeatureForceData_u1);
VARIANT spData_u1;
pFeatureForceData_u1-
>get_FunctionInterpolatedValues(&spData_u1);
SafeDoubleArray my_u1(spData_u1);
for (int idx = 0; idx < k; idx++)
{
my_u1[idx] = Time[idx];
my_u1[idx + k] = u1[idx];
};
pFeatureForceData_u1-
>put_FunctionInterpolatedValues(spData_u1);

```

a)

```

pMotionStudyResults->GetCMPPosition(Time[k],
Comp, &pCMPPosition);
pCMPPosition->QueryInterface(IID_IMathVector,
(LPVOID*)&pMathVector);
double CMPPosition[3];
double LengthCMPPosition;
pMathVector->GetLength(&LengthCMPPosition);
VARIANT ArrayDataOutCMP;
pMathVector->get_ArrayData(&ArrayDataOutCMP);
SafeDoubleArray
my_ArrayDataOutCMP(ArrayDataOutCMP);

```

b)

Р и с. 10. Программный доступ к данным SolidWorks Motion:
a) определение управления, б) импорт результатов

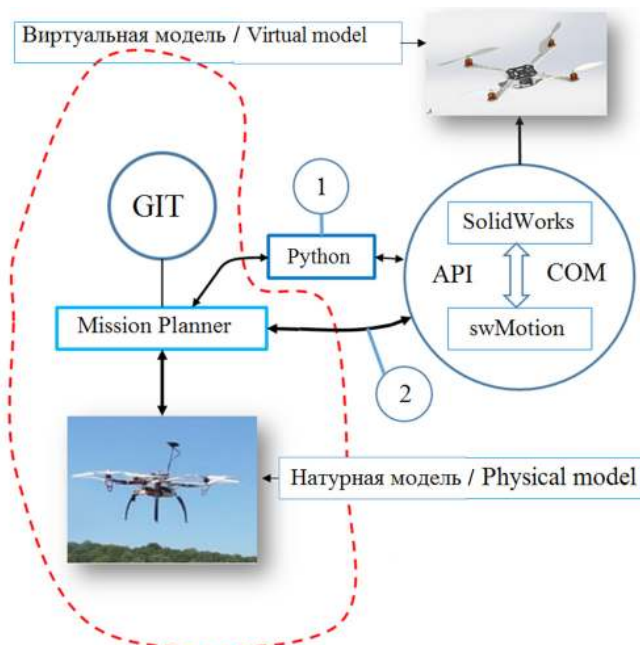
F i g. 10. Program access to SolidWorks Motion data: a) control definition, b) results import

дели¹⁸. В данной статье реализован способ, отмеченный на рисунке цифрой 1. Программное обеспечение swMotion не только планирует траекторию и реализует управление полетом в виртуальной среде, но и создает файл WayPoint в формате Mission Planner. Этот файл задает желаемую траекторию, а Mission Planner реализует управление полетом натурной модели по этой траектории (рис. 5–8).

Программа управления может быть загружена в полетный контроллер как непосредственно программой Mission Planner, так и с использованием скриптов Python¹⁹. Существенным недостатком данного подхода являются сложности создания двунаправленных ассоциатив-

ных связей между натурной и виртуальной моделями. В частности, swMotion должен получать доступ к телеметрической информации, передаваемой в MissionPlanner системой GPS натурной модели.

Проблема может быть решена на основе метода, помеченного на рис. 11 цифрой 2. В этом случае Mission Planner используется как программное обеспечение с открытым кодом GIT (Global Information Tracker, распределенной системы контроля версий), подлежит компиляции, а программные модули swMotion встраиваются в MissionPlanner. Именно такой подход авторы намерены реализовать в перспективе²⁰.



Р и с. 11. Интегрированная модель

F i g. 11. The integrated model

¹⁸ 3D-Fly.mp4 – Google Диск. URL: <https://drive.google.com/file/d/1p4WmgRDwRewiWCWU4NYax6egGQhO9OoK/view?usp=sharing>; RealFly.mp4 – Google Диск. URL: <https://drive.google.com/file/d/1mUWgTT4QbrBvPbVzLAL5exsBplvIhI0o/view?usp=sharing>

¹⁹ Using Python Scripts in Mission Planner – Mission Planner documentation. URL: <http://ardupilot.org/planner/docs/using-python-scripts-in-mission-planner.html>

²⁰ GitHub – ArduPilot/ardupilot: ArduPlane, ArduCopter, ArduRover source. URL: <https://github.com/ArduPilot/ardupilot>

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. **Макаров И. М., Рахманкулов В. З., Ахрем А. А.** Виртуальное моделирование и интеллектуальное управление сложными компьютерно-интегрированными системами // Информационные технологии и вычислительные системы. 2007. № 2. С. 11–24. URL: http://www.jitcs.ru/index.php?option=com_content&view=article&id=179
2. Развитие технологии интеллектуального управления для создания перспективных образцов ВВТ на базе новых средств комплексной автоматизации проектирования / И. М. Макаров [и др.] // Известия ЮФУ. Технические науки. 2013. № 3. С. 7–14. URL: <http://old.izv-tn.tti.sfedu.ru/?p=1565>
3. **Чугунов М. В., Полунина И. Н.** Междисциплинарное моделирование роботов с использованием систем автоматизированного проектирования // Вестник Мордовского университета. 2018. Т. 28, № 2. С. 181–190. DOI: <https://doi.org/10.15507/0236-2910.028.201802.181-190>
4. **Orsag M., Bogdan S.** Influence of forward and descent flight on quadrotor dynamics // Recent Advances in Aircraft Technology / Ed. R. Agarwal. Zagreb : InTech, 2012. P. 141–156. DOI: <https://doi.org/10.5772/37438>
5. **Гэн К., Чулин Н. А.** Алгоритмы стабилизации для автоматического управления траекторным движением квадрокоптера // Наука и образование. МГТУ им. Н. Э. Баумана. 2015. № 5. С. 218–235. URL: <http://engineering-science.ru/doc/771076.html>
6. **Петров В. Ф., Барунин А. А., Терентьев А. И.** Модель системы автоматического управления беспилотным летательным аппаратом // Известия Тульского государственного университета. Технические науки. 2014. № 12-2. С. 217–225. URL: https://tidings.tsu.tula.ru/tidings/pdf/web/preview_therest_ru.php?x=tsu_izv_technical_sciences_2014_12_part_2&year=2014
7. **Kotarski D., Benić Z., Krzmar M.** Control design for unmanned aerial vehicles with four rotors // Interdisciplinary Description of Complex Systems. 2016. Vol. 14, no. 2. P. 236–245. URL: <https://hrcak.srce.hr/154449>
8. **Белявский А. О., Томашевич С. И.** Синтез адаптивной системы управления квадрокоптером методом пассивации // Управление большими системами. 2016. Вып. 63. С. 155–181. URL: http://ubs.mtas.ru/search/search_results_ubs_new.php?publication_id=21495&IBLOCK_ID=20
9. **Ермаченков Д. И., Фазли Т. Г. К., Петренко Е. О.** Разработка конструкции рамы квадрокоптера для удаленного мониторинга объектов // Интернет-журнал «Науковедение». 2016. Т. 8, № 6. С. 45. DOI: <http://dx.doi.org/10.15862/45TVN616>
10. **Ермаченков Д. И., Фазли Т. К.** Плата управления квадрокоптера для удаленного мониторинга объектов // Инженерный вестник. 2016. № 8. С. 12–27. URL: <http://ainjournal.ru/doc/847059.html>
11. **Гоголев А. А.** Полунатурное моделирование беспилотных летательных аппаратов типа мультикоптер // Труды МАИ. 2017. № 92. С. 29. URL: <http://trudymai.ru/published.php?ID=77238>
12. **Shaqura M., Shamma J. S.** An automated quadcopter CAD based design and modeling platform using Solidworks API and smart dynamic assembly // Proceedings of the 14th International Conference on Informatics in Control, Automation and Robotics. 2017. Vol. 2. P. 122–131. DOI: <https://doi.org/10.5220/0006438601220131>
13. **Попков М. А., Чугунов М. В.** Проектирование и оптимизация несущей системы квадрокоптера // Молодой ученый. 2018. № 14 (200). С. 30–35. URL: <https://moluch.ru/archive/200/49131>
14. **Mellinger D., Kumar V.** Minimum snap trajectory generation and control for quadrotors // 2011 IEEE International Conference on Robotics and Automation. 2011. P. 2520–2525. DOI: <https://doi.org/10.1109/ICRA.2011.5980409>
15. **Кочкаров А. А., Агишев Р. Т.** Сравнительный анализ полетов квадрокоптера вдоль траекторий различной степени гладкости // Современная наука. 2016. № 3. С. 17–22. URL: http://www.sovnauka.com/ru/articles/2016-3/sravnitelnyj_analiz_poletov_kvadrokoptera_vdol_traektorij_razlichnoj_stepeni_gladkosti
16. **Piegl L., Tiller W.** The NURBS Book. 2nd ed. Berlin ; Heidelberg : Springer-Verlag, 1997. 646 p. DOI: <https://doi.org/10.1007/978-3-642-59223-2>

17. Direct method based control system for an autonomous quadrotor / I. D. Cowling [et al.] // Journal of Intelligent & Robotic Systems. 2010. Vol. 60, issue 2. P. 285–316. DOI: <https://doi.org/10.1007/s10846-010-9416-9>
18. The GRASP multiple micro-UAV testbed / N. Michael [et al.] // IEEE Robotics & Automation Magazine. 2010. Vol. 17, issue 3. P. 56–65. DOI: <https://doi.org/10.1109/MRA.2010.937855>
19. **Попков М. А., Чугунов М. В.** Моделирование полета квадрокоптера в среде SolidWorks Motion // Молодой ученый. 2018. № 16 (202). С. 135–138. URL: <https://moluch.ru/archive/202/49502>
20. Анализ и проектирование несущих элементов конструкций подвижного состава / М. В. Чугунов [и др.] // Наука и образование. МГТУ им. Н. Э. Баумана. 2014. № 9. С. 216–226. URL: <http://engineering-science.ru/doc/726307.html>

Поступила 01.11.2018; принята к публикации 11.01.2019; опубликована онлайн 28.06.2019

Об авторах:

Чугунов Михаил Владимирович, заведующий, кафедра конструкторско-технологической информатики, Рузаевский институт машиностроения, ФГБОУ ВО «Мордовский государственный университет им. Н. П. Огарёва» (431440, Россия, г. Рузаевка, ул. Ленина, д. 93), кандидат технических наук, доцент, ResearcherID: H-7452-2018, ORCID: <https://orcid.org/0000-0001-5318-5684>, m.v.chugunov@mail.ru

Полунина Ирина Николаевна, доцент, кафедра конструкторско-технологической информатики, Рузаевский институт машиностроения, ФГБОУ ВО «Мордовский государственный университет им. Н. П. Огарёва» (431440, Россия, г. Рузаевка, ул. Ленина, д. 93), кандидат педагогических наук, ResearcherID: H-7473-2018, ORCID: <https://orcid.org/0000-0002-1093-8401>, my_pk@mail.ru

Попков Михаил Андреевич, студент, кафедра конструкторско-технологической информатики, Рузаевский институт машиностроения, ФГБОУ ВО «Мордовский государственный университет им. Н. П. Огарёва» (431440, Россия, г. Рузаевка, ул. Ленина, д. 93), ResearcherID: F-5990-2019, ORCID: <https://orcid.org/0000-0001-7422-9076>, m.a.popkov2016@mail.ru

Заявленный вклад соавторов:

М. В. Чугунов – разработка методики и программного обеспечения для решения задач построения интегрированных моделей квадрокоптера; И. Н. Полунина – расчетные процедуры, компьютерное редактирование текста и графики; М. А. Попков – построение 3D моделей.

Все авторы прочитали и одобрили окончательный вариант рукописи.

REFERENCES

1. Makarov I.M., Rakhmankulov V.Z., Akhrem A.A. Virtual modeling and intellectual management of the complex computer integrated systems. *Informatsionnyye tekhnologii i vychislitelnyye sistemy* = Journal of Information Technologies and Computing Systems. 2007; 2:11-24. Available at: http://www.jitcs.ru/index.php?option=com_content&view=article&id=179 (In Russ.)
2. Makarov I.M., Lokhin V.M., Manko S.V., Romanov M.P., Aleksandrova R.I. Development of intelligent control technology for creation of autonomous objects on the basis of complex automation design. *Izvestiya YuFU. Tekhnicheskkiye nauki* = Izvestiya SFedU. Engineering Sciences. 2013; 3:7-14. Available at: <http://old.izv-tn.tti.sfedu.ru/?p=1565> (In Russ.)
3. Chugunov M.V., Polunina I.N. Interdisciplinary modelling of robots using CAD/CAE technology. *Vestnik Mordovskogo universiteta* = Mordovia University Bulletin. 2018; 28(2):181-190. DOI: <https://doi.org/10.15507/0236-2910.028.201802.181-190> (In Russ.)

4. Orsag M., Bogdan S. Influence of forward and descent flight on quadrotor dynamics. In: Agarwal R. (ed.) *Recent Advances in Aircraft Technology*. Zagreb: InTech, 2012. p. 141-156. DOI: <https://doi.org/10.5772/37438>
5. Gen K., Chulin N.A. Stabilization algorithms for automatic control of the trajectory movement of quadcopter. *Nauka i obrazovaniye. MG TU im. N. E. Baumana* = Science and Education of the Bauman MSTU. 2015; 5:218-235. Available at: <http://engineering-science.ru/doc/771076.html> (In Russ.)
6. Petrov V.F., Barunin A.A., Terentiev A.I. Model of the automatic control system unmanned aerial vehicle. *Izvestiya Tul'skogo gosudarstvennogo universiteta. Tekhnicheskiye nauki* = News of the Tula State University. Technical Sciences. 2014; 12(2):217-225. Available at: https://tidings.tsu.tula.ru/tidings/pdf/web/preview_therest_ru.php?x=tsu_izv_technical_sciences_2014_12_part_2&year=2014 (In Russ.)
7. Kotarski D., Benić Z., Krznar M. Control design for unmanned aerial vehicles with four rotors. *Interdisciplinary Description of Complex Systems : INDECS*. 2016; 14(2):236-245. Available at: <https://hrcak.srce.hr/154449> (In Russ.)
8. Belyavsky A.O., Tomashevich S.I. Passivity-based method for quadrotor control. *Upravleniye bolshimi sistemami* = Large-Scale Systems Control. 2016; 63:155-181. Available at: http://ubs.mtas.ru/search/search_results_ubs_new.php?publication_id=21495&BLOCK_ID=20 (In Russ.)
9. Ermachenkov D.I., Fazly T.G.K., Petrenko E.O. Quadrotor frame design for remote object monitoring. *Naukovedeniye* = Science Studies. 2016; 8(6):45. DOI: <http://dx.doi.org/10.15862/45TVN616> (In Russ.)
10. Ermachenkov D.I., Fazly T.K. Chip for management of quadcopter for remote monitoring of objects. *Inzhenernyy vestnik* = Engineering Bulletin. 2016; 8:12-27. Available at: <http://ainjournal.ru/doc/847059.html> (In Russ.)
11. Gogolev A.A. Semi-natural modelling of unmanned aerial vehicles like multicopter. *Trudy MAI* = Works of MAI. 2017; 92:29. Available at: <http://trudymai.ru/published.php?ID=77238> (In Russ.)
12. Shaqura M., Shamma J.S. An automated quadcopter CAD based design and modeling platform using solidworks API and smart dynamic assembly. *Proceedings of the 14th International Conference on Informatics in Control, Automation and Robotics*. 2017; 2:122-131. DOI: <https://doi.org/10.5220/0006438601220131>
13. Popkov M.A., Chugunov V.V. Design and optimization of frame system of quadcopter. *Molodoy uchenyy* = Young Scientist. 2018; 14:30-35. Available at: <http://moluch.ru/archive/200/49131> (In Russ.)
14. Mellinger D., Kumar V. Minimum snap trajectory generation and control for quadrotors. In: *2011 IEEE International Conference on Robotics and Automation*. 2011. p. 2520-2525. DOI: <https://doi.org/10.1109/ICRA.2011.5980409>
15. Kochkarov A.A., Agishev R.T. Comparative analysis of the flights of a quadrotor along trajectories of various degrees of smoothness. *Modern Science*. 2016; 3:17-22. Available at: http://www.sovnauka.com/ru/articles/2016-3/sravnitelnyj_analiz_poletov_kvadroptera_vdol_traektorij_razlichnoj_stepeni_gladkosti (In Russ.)
16. Piegel L., Tiller W. *The NURBS Book*. 2nd ed. Berlin ; Heidelberg : Springer-Verlag, 1997. DOI: <https://doi.org/10.1007/978-3-642-59223-2>
17. Cowling I.D., Yakimenko O.A., Whidborne J.F., Cooke A.K. Direct method based control system for an autonomous quadrotor. *Journal of Intelligent & Robotic Systems*. 2010; 60(2):285-316. DOI: <https://doi.org/10.1007/s10846-010-9416-9>
18. Michael N., Mellinger D., Lindsey Q., Kumar V. The GRASP multiple micro-UAV testbed. *IEEE Robotics & Automation Magazine*. 2010; 17(3):56-65. DOI: <https://doi.org/10.1109/MRA.2010.937855>
19. Popkov M.A., Chugunov M.V. Modeling of flight of quadcopter in Solidworks Motion software. *Molodoy uchenyy* = Young Scientist. 2018; 16:135-138. Available at: <http://moluch.ru/archive/202/49502> (In Russ.)
20. Chugunov M.V., Osyka V.V., Kudaev S.P., Kuzmichyov N.D., Klyomin V.N. Analysis and design of rolling stock elements. *Nauka i obrazovaniye. MG TU im. N. E. Baumana* = Science & Education of the Bauman MSTU. 2014; 9:216-226. Available at: <http://engineering-science.ru/doc/726307.html> (In Russ.)

Received 01.11.2018; revised 11.01.2019; published online 28.06.2019

About authors:

Mikhail V. Chugunov, Head, Chair of Design and Technology Informatics, Ruzaevka Institute of Mechanical Engineering, National Research Mordovia State University (93 Lenina St., Ruzaevka 431440, Russia), Ph.D. (Engineering), Associate Professor, ResearcherID: H-7452-2018, ORCID: <https://orcid.org/0000-0001-5318-5684>, m.v.chugunov@mail.ru

Irina N. Polunina, Associate Professor, Chair of Design and Technology Informatics, Ruzaevka Institute of Mechanical Engineering, National Research Mordovia State University (93 Lenina St., Ruzaevka 431440, Russia), Ph.D. (Pedagogy), ResearcherID: H-7473-2018, ORCID: <https://orcid.org/0000-0002-1093-8401>, my_pk@mail.ru

Mikhail A. Popkov, Student, Chair of Design and Technology Informatics, Ruzaevka Institute of Mechanical Engineering, National Research Mordovia State University (93 Lenina St., Ruzaevka 431440, Russia), ResearcherID: F-5990-2019, ORCID: <https://orcid.org/0000-0001-7422-9076>, m.a.popkov2016@mail.ru

Contribution of the authors:

M. V. Chugunov – the development of the technique and software for integrated quadcopter systems building; I. N. Polunina – the computer analysis of procedures, computer edition of the text and graphics; M. A. Popkov – 3D-model design.

All authors have read and approved the final version of the paper.