

ИСПОЛЬЗОВАНИЕ ПАКЕТА MATLAB.SIMULINK ПРИ БАЛЛИСТИЧЕСКИХ РАСЧЕТАХ

М.Д. Никанорова

rnikanorova@gmail.com

И.В. Веденичев

Vedenicheviv@gmail.com

МГТУ им. Н.Э. Баумана, Москва, Российская Федерация

Аннотация

Ключевые слова

Проведено моделирование полета материальной точки в среде Matlab.Simulink на языке C++. Проанализированы полученные результаты. Выявлены преимущества использования данного пакета.

Проведено моделирование полета идеально наводящейся зенитной управляемой ракеты и реального полета зенитной управляемой ракеты. Доказана невозможность использования системы наведения без корректирующих фильтров.

Matlab.Simulink, баллистика, численное моделирование, корректирующие фильтры

Поступила в редакцию 12.06.2017

© МГТУ им. Н.Э. Баумана, 2017

Использование современной цифровой вычислительной техники позволяет произвести вычисление большого объема информации быстро и с необходимой точностью. Поэтому в современной практике баллистических расчетов активно используется численное моделирование движения летательного аппарата (ЛА), которое отражает реальный полет с определенными допущениями [1].

Различные прикладные пакеты, используемые для исследования и проектирования электромеханических систем, позволили качественно изменить и существенно расширить возможности инженера. Система Matlab (матричная лаборатория) была создана специалистами фирмы MathWorks, Inc. как язык программирования высокого уровня для технических вычислений. Особенно тщательно в Matlab проработаны алгоритмы матричных операций, лежащие в основе большинства средств моделирования сложных систем. Система Matlab имеет открытую архитектуру, что дает полный доступ пользователям к ее кодам на гибком и мощном (и в то же время простом) языке программирования этой системы.

Пакет Simulink является приложением к пакету Matlab. В некотором смысле Simulink является самостоятельным продуктом фирмы MathWorks. Однако работает Simulink только при наличии ядра Matlab и использует функции, входящие в его состав. Пакет расширений Simulink предназначен для моделирования динамических систем, модели которых составляются из отдельных блоков. Этот пакет является наиболее востребованным из программ, созданных на основе системы Matlab. В нем реализованы принципы визуально-ориентированного программирования, что позволяет легко набирать нужные блоки и соединять их с целью составления модели системы или устройства. При этом сложнейшие

уравнения состояния, описывающие работу моделей систем или устройств, формируются автоматически.

Удобство графического пользовательского интерфейса, обилие различных блоков во множестве библиотек, разнообразие виртуальных средств регистрации и визуализации результатов моделирования позволяет использовать пакет Matlab.Simulink для баллистических расчетов различной сложности [2].

Моделирование полета материальной точки. Для демонстрации простоты и легкости построения различных систем в пакете Matlab.Simulink проведем сравнительный анализ решения одной и той же задачи в рассматриваемом пакете и на языке C++.

В качестве примера рассмотрим задачу о вылете пули из ствола пневматической винтовки. Исходными данными являются параметры пневматической винтовки, определенные при проведении эксперимента на баллистической трассе в лаборатории кафедры «Динамика и управление полетом ракет и космических аппаратов» МГТУ им. Н.Э. Баумана. Баллистическая трасса — это специально оборудованная стрелковая трасса, позволяющая в полигонных или лабораторных условиях производить стрельбу специальными моделями, пулями, снарядами для исследования процесса свободного полёта ЛА и экспериментального определения [3].

Начальные условия принимаем следующие: угол бросания $\theta = 0,00822$ рад; масса $m_{cp} = 0,000488$ кг; начальная скорость $V_{cp} = 178,14$ м/с; длина баллистической трассы $l = 33,8$ м.

Далее запишем математическую модель [4]:

$$\left\{ \begin{array}{l} \frac{dx}{dt} = V \cos \theta; \\ \frac{dy}{dt} = V \sin \theta; \\ m \frac{dV}{dt} = -X_a - G \sin \theta; \\ m V \frac{d\theta}{dt} = -G \cos \theta, \end{array} \right.$$

где G — сила тяжести, $G = mg$; g — ускорение свободного падения, $g = 9,80665 \frac{M}{c}$; X_a — сила лобового сопротивления, $X_a = C_{X_a} q S$; q — скорост-

ной напор, $q = \frac{\rho(y) V^2(t)}{2}$.

Сначала реализуем данную модель на языке C++. Для написания программы необходимо подключить некоторое число библиотек, зависящее от выбранного метода решения, затем объявить и присвоить тип каждой из большого числа используемых переменных. Далее для удобства вывести полученные в ходе решения данные в файл и произвести интегрирование выбранным методом. Все это требует определенных навыков работы с выбранным языком, большого

количества времени для записи программы, компилирование и устранение ошибок, появляющихся в процессе решения.

Обычно полученная программа занимает от 200 до 300 строк и много времени.

Численное интегрирование производилось методом Рунге—Кутты четвертого порядка, а программирование — в среде MCSVisualStudio. Результаты расчетов представлены на рис. 1–3.

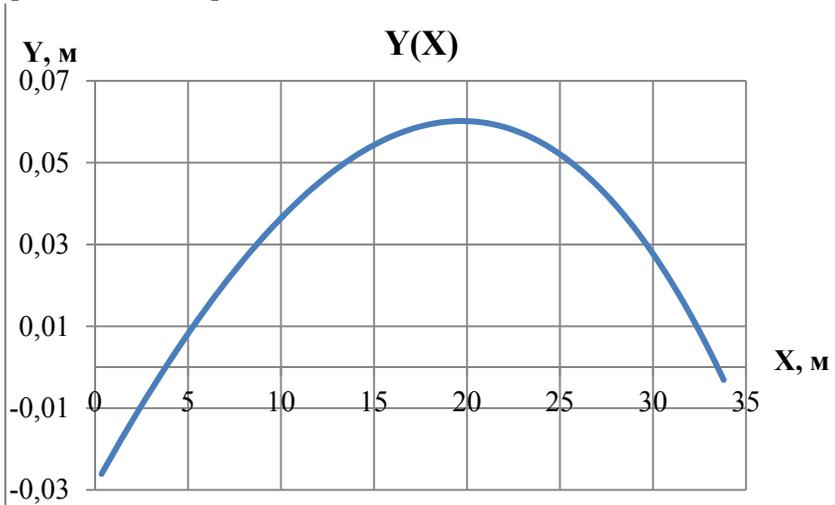


Рис. 1. Траектория движения ЛА

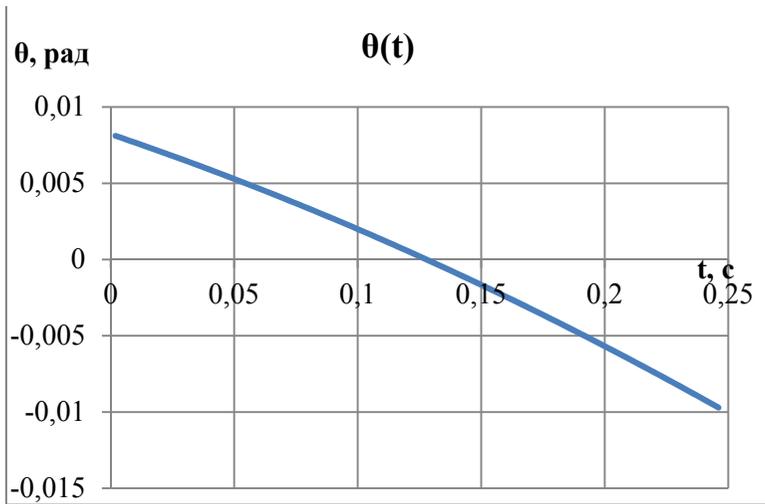


Рис. 2. Зависимость угла бросания от времени

В противовес языку C++ среда Matlab.Simulink проста в использовании, не требует большого количества времени на изучение принципов построения модели, а библиотеки предоставляют широкие возможности для решения задач различных областей программирования. Например, для численного интегрирования программы на языке C++ необходимо написать процедуру, занимающую

большое количество строк и каждый раз при необходимости вызывать ее, тогда как в среде Matlab.Simulink интегрирование является стандартной процедурой, включающей в себя различные методы, реализующие необходимую точность расчета. Имеется возможность выбрать метод как с фиксированным шагом (например, метод Эйлера, Рунге—Кутты четвертого порядка), так и с переменным (метод Адамса). Интегрирование заданной системы проводилось методом Эйлера первого порядка с шагом 0,001 с [5].

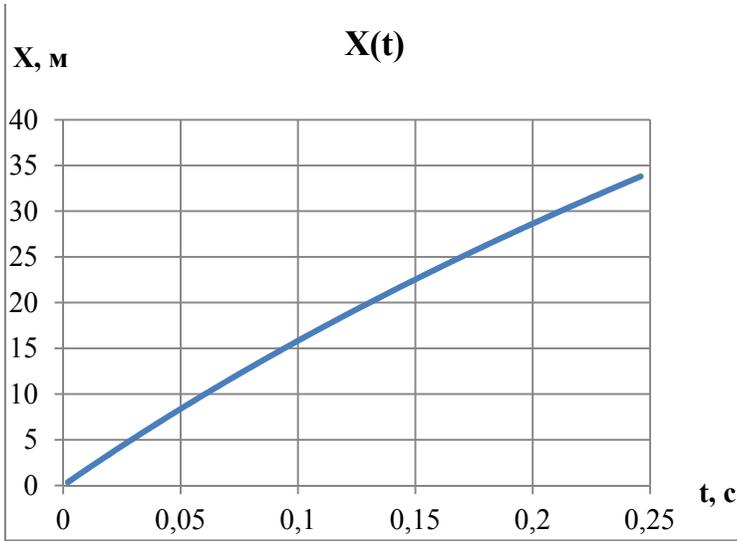


Рис. 3. Зависимость дальности от времени

Для решения задачи о полете тела, брошенного под углом к горизонту, будем использовать блоки библиотеки AerospaceBlockset, предназначенные именно для решения задач полета.

Первый необходимый нам блок — это 4th OrderPointMass (Longitudinal). Этот блок выполняет вычисление поступательного движения материальной точки (МТ). В нем задаются начальные угол наклона траектории, скорость, дальность и высота. Входными величинами являются силы F_x и F_z . Система координат, в которой выполняется вычисление, выглядит следующим образом: ось X_a сонаправлена вектору скорости движения тела в воздушном потоке, а ось Y_a направлена вверх перпендикулярно оси X_a . Для задания сил F_x и F_z данного блока используется блок 4thOrderPointMassForces (Longitudinal) из библиотеки AerospaceBlockset. Обнулим входные воздействия для не используемых в данной задаче величин (тяги, подъемной силы, углов крена и атаки) и получится, что нам необходимы только значения угла наклона траектории и силы тяжести, которые нам известны (рис. 4).

Следующий блок — это COESA (Committee On Extension to the Standard Atmosphere) Atmosphere Model. С помощью этого блока вычисляются параметры атмосферы (T , a , p , ρ) на основе стандарта США 1976 года.

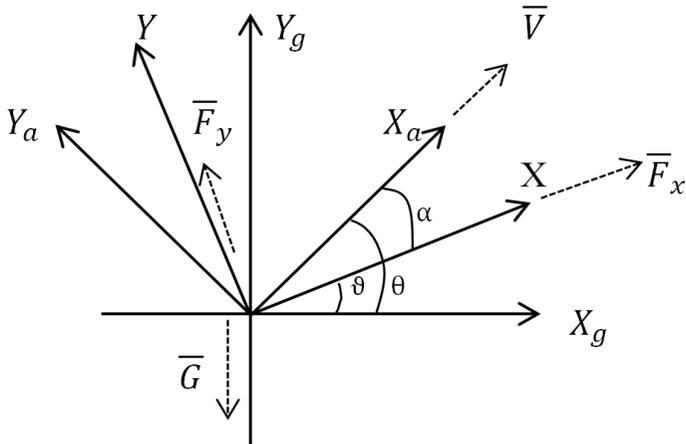


Рис. 4. Система координат блока 4th Order Point Mass Forces (Longitudinal)

Модель данной задачи, построенная с использованием функциональных блоков среды Matlab.Simulink, представлена на рис. 5.

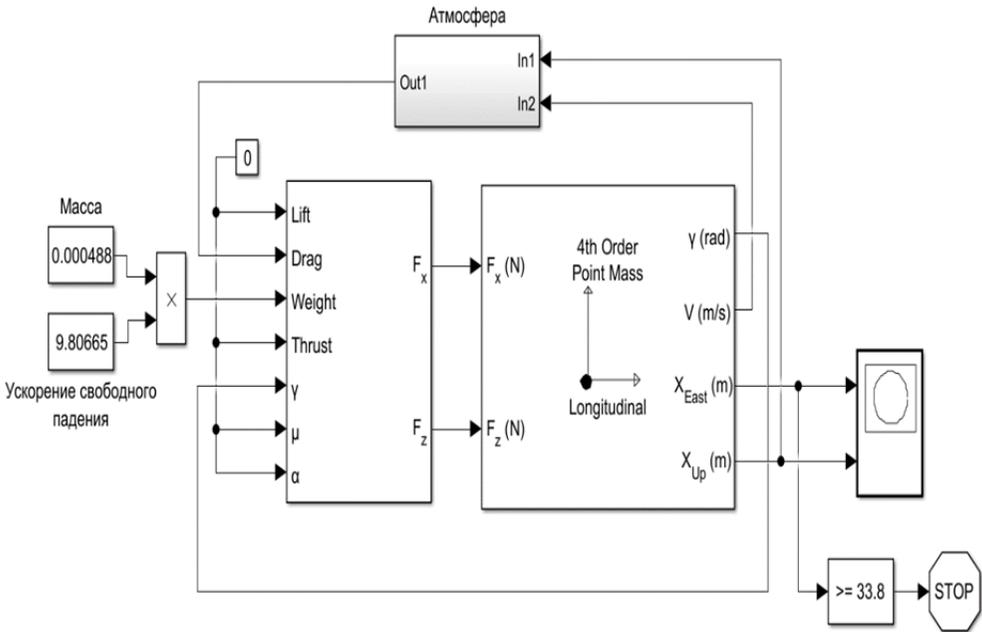


Рис. 5. Модель системы в блоках Matlab.Simulink

Для проверки правильности решения блоками стандартной библиотеки было также проведено моделирование с заданием проекций сил через угол наклона траектории.

В состав приведенной выше системы также входит подсистема «Атмосфера», вид которой представлен ниже (рис. 6). Эта подсистема включает в себя блок, отвечающий за модель атмосферы из стандартной библиотеки.

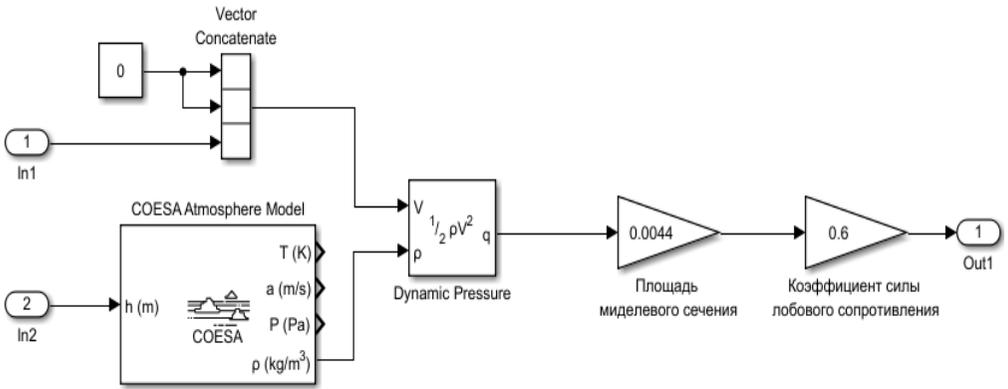


Рис. 6. Подсистема «Атмосфера»

В состав представленной выше системы также входит блок VectorConcatenate, который необходим для того, чтобы преобразовать скалярное значение скорости, полученное на выходе из блока 4th OrderPointMass (Longitudinal), в $V = (V; 0; 0)$, так как в блоке скоростного напора скорость должна представляться в векторном формате. Результаты решения представлены на рис. 7–9.

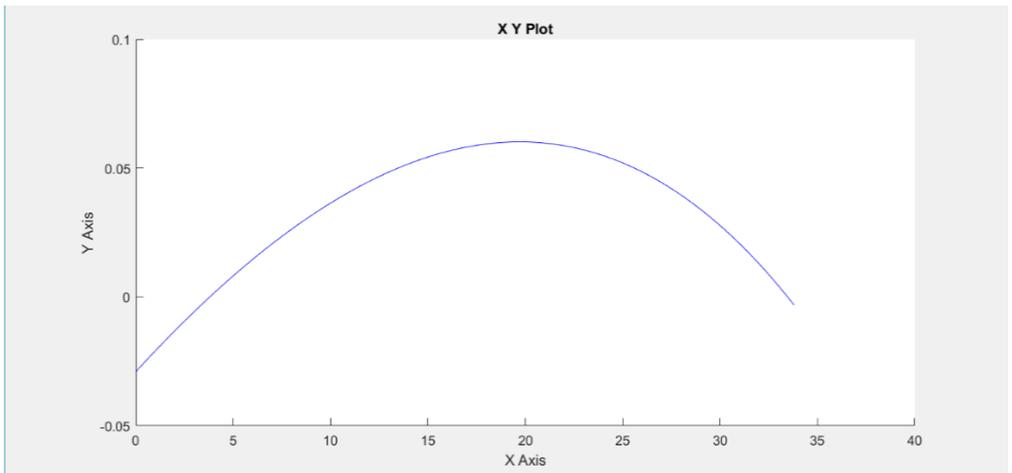


Рис. 7. Траектория полета ЛА в атмосфере

Для удобства сравнения значений величин, полученных при решении задачи на двух различных языках программирования, приведем значения этих данных в конечный момент времени (см. табл.).

Из таблицы видно, что конечные результаты эквивалентны, а небольшая разница объясняется использованием различных стандартов атмосферы. Полученные данные позволяют судить об идентичности и взаимозаменяемости решений, полученных в Matlab.Simulink и на языке C++. Однако очевидно, что решение в Matlab.Simulink более наглядное и простое, применительно к моделированию задач баллистики.

Значения параметров в конечный момент времени

Параметр	Результаты, полученные в Matlab.Simulink	Результаты, полученные в C++
Момент времени t , с	0,246	0,246
Дальность X , м	33,8053	33,8055
Высота Y , м	-0,00305	-0,00304
Скорость V , м/с	108,0	108,0
Угол наклона траектории γ , рад	-0,0097	-0,0097

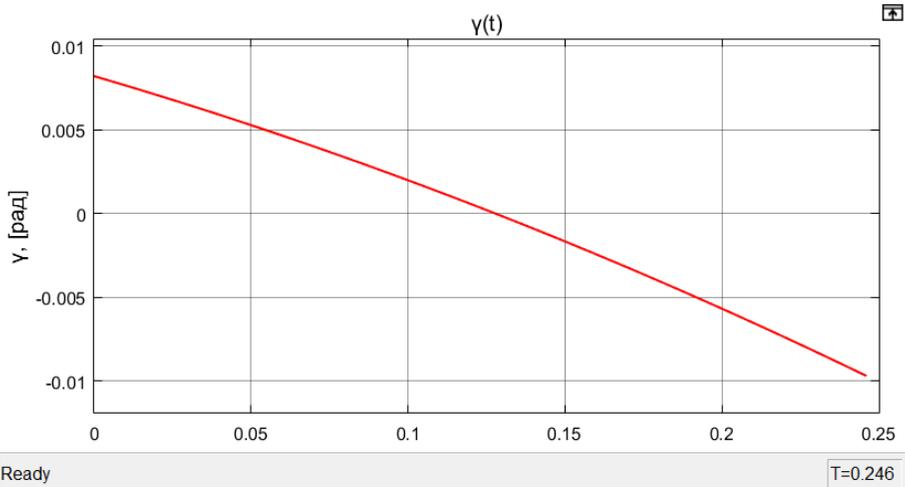


Рис. 8. Зависимость угла наклона траектории от времени полета ЛА в атмосфере

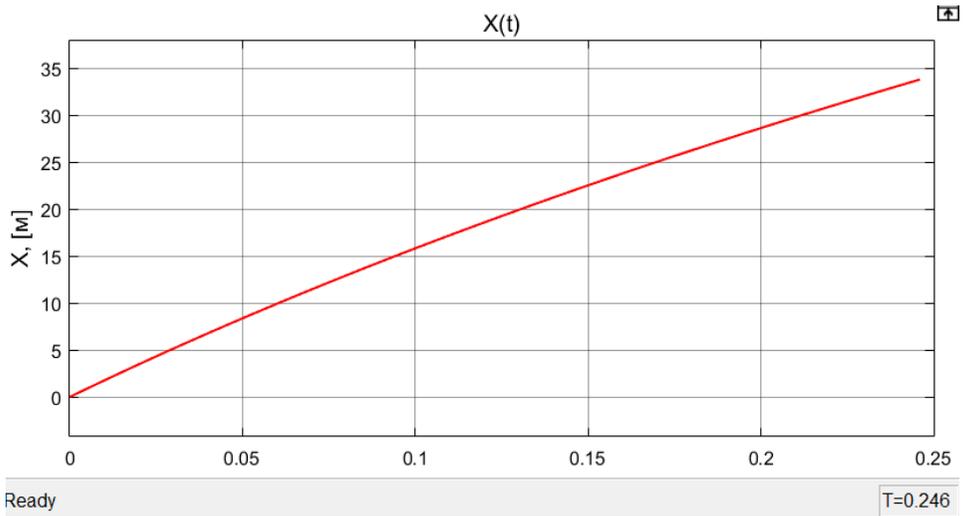


Рис. 9. Зависимость дальности от времени полета ЛА в атмосфере

Решение задачи построения модели идеального и реального наведения зенитной управляемой ракеты (ЗУР). После того как мы убедились в простоте построения элементарных баллистических моделей в среде Matlab.Simulink приступим к решению более сложных задач. Произведем моделирование полета идеаль-

но наводящейся ЗУР. Под идеальным будем понимать такой вариант наведения, при котором органы управления идеально отрабатывают требуемые сигналы, а сама ЗУР мгновенно принимает необходимое положение. В качестве метода наведения был выбран метод наведения по трем точкам. Суть его заключается в том, чтобы удерживать ракету на линии, соединяющей командный пункт с целью, как показано на рис. 10. Выбор данного метода обусловлен простотой его реализации. Также для упрощения будем решать плоскую задачу.

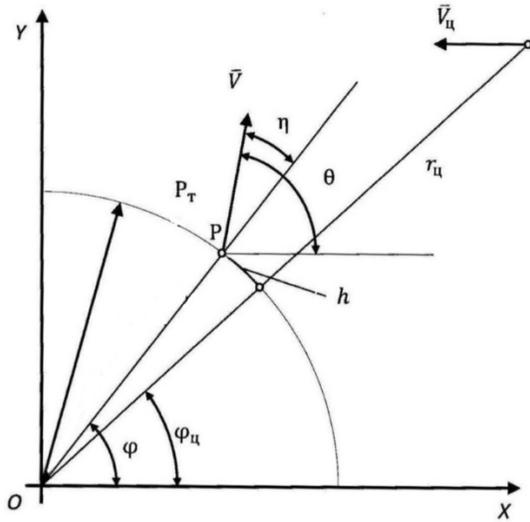


Рис. 10. Метод наведения по трем точкам:

φ — угол визирования ракеты; φ_c — угол визирования цели; h — текущий промах; θ — угол наклона траектории ракеты; r_c — расстояние от командного пункта до цели; $\eta = \theta - \varphi$

На рисунке 11 изображена модель, описывающая движение ЗУР, в которой можно выделить несколько основных блоков.

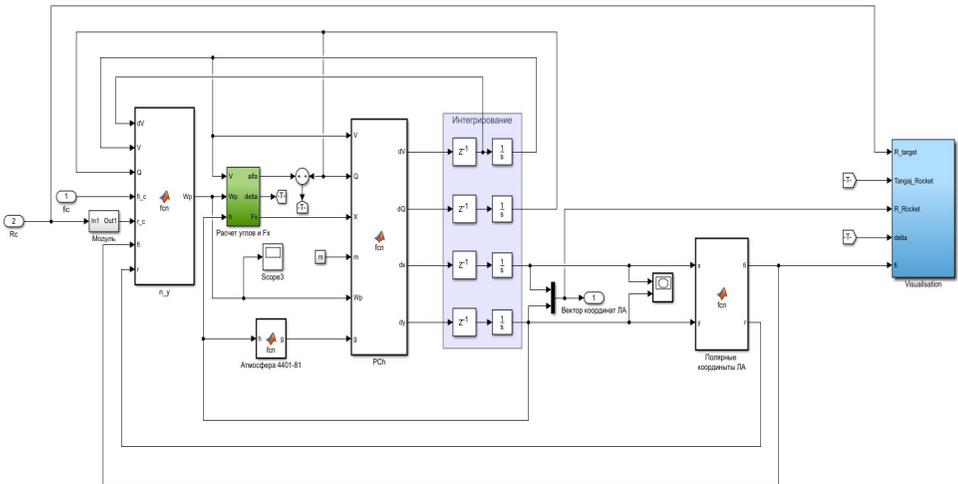


Рис. 11. Модель, описывающая движение ЗУР

Блок расчета требуемого нормального ускорения ЗУР от поверхностных сил. Математическая модель выглядит следующим образом:

$$\begin{cases} r_p \dot{\varphi}_k = V_p \sin(\theta - \varphi_k); \\ \dot{r}_p = V_p \cos(\theta - \varphi_k); \\ V_p \dot{\theta} = W_p - g \cos(\theta). \end{cases}$$

С помощью этой системы находят требуемое нормальное ускорение.

Блок расчета угла отклонения рулей, требуемого угла атаки и продольной силы F_x . Исходя из значения требуемого нормального ускорения, рассчитывают требуемый угол атаки, затем — стабилизирующий его угол отклонения рулей. Аэродинамические коэффициенты заданы в виде интерполяционной таблицы.

Блок расчета правых частей. Математическая модель описывается таким образом [6]:

$$\begin{cases} \frac{dV}{dt} = \frac{P - F_x}{m} - g \sin(\theta); \\ \frac{d\theta}{dt} = \frac{W_p}{V} - g \frac{\cos(\theta)}{V}; \\ \frac{dx}{dt} = V \cos(\theta); \\ \frac{dy}{dt} = V \sin(\theta). \end{cases}$$

Блок интегрирования методом Рунге—Кутты четвертого порядка с шагом 0,001 с.

Блок визуализации, написанный стандартными средствами Matlab.Simulink. При этом визуализация может проводиться сторонними средствами.

Цель движется равномерно, параллельно земле. Параметры полета ЗУР при идеальном наведении на цель, обнаруженную в точке (30000; 10000), приведены на рис. 12 и 13.

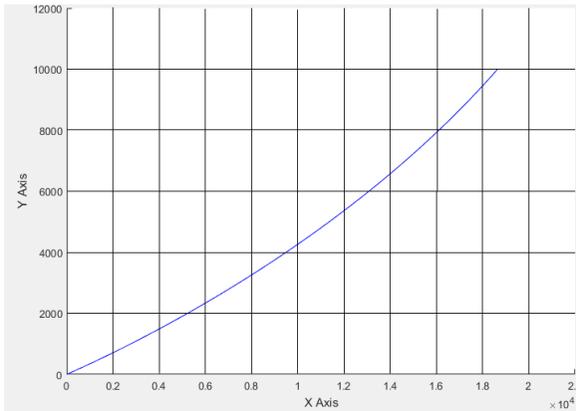
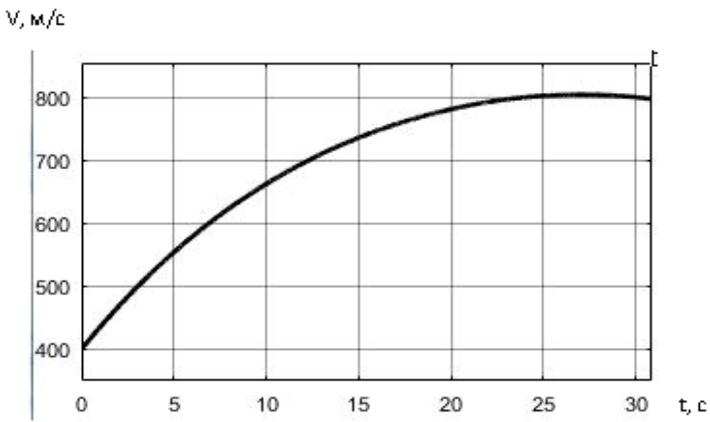
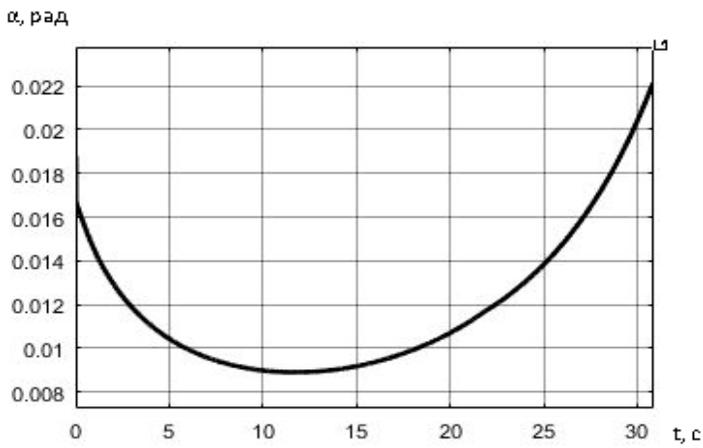


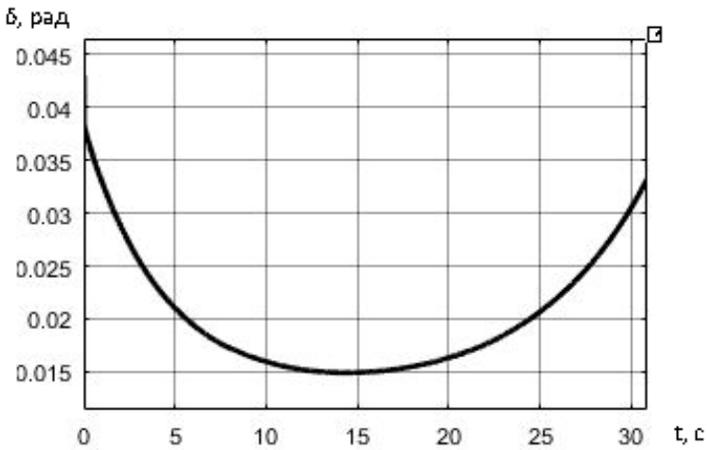
Рис. 12. Траектория полета ЗУР при идеальном наведении



a



б



в

Рис. 13. Изменение скорости (*a*), угла атаки (*б*), угла отклонения рулей (*в*) при идеальном наведении

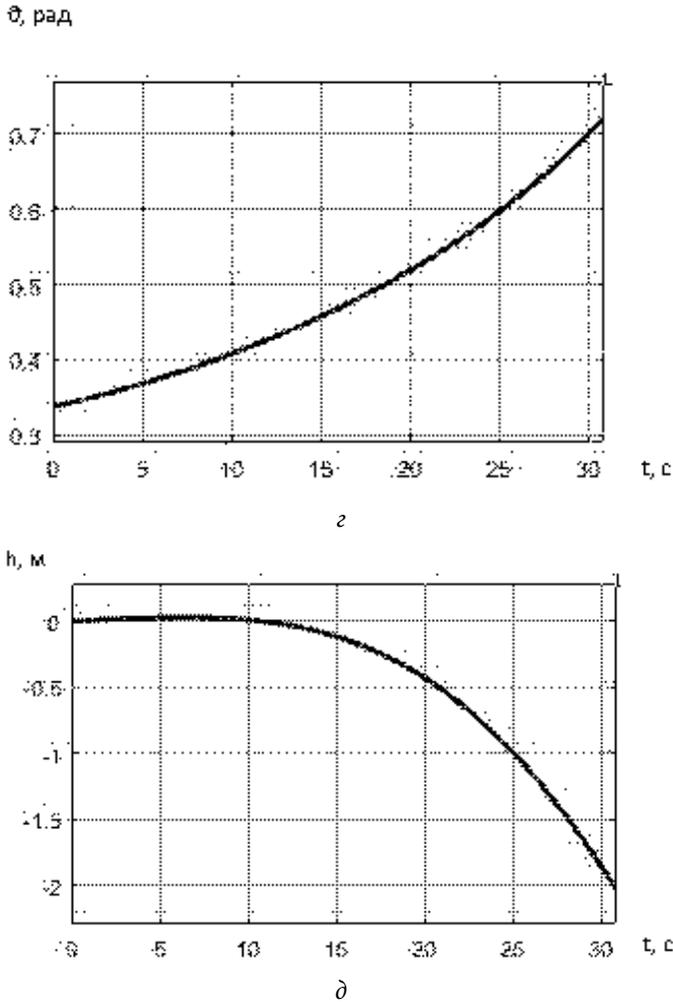


Рис. 13. Изменение угла тангажа (z) и промах (δ) ЗУР при идеальном наведении

При данном шаге промах составил 2,1 м. Стоит отметить, что при моделировании идеального наведения значение промаха зависит только от шага интегрирования.

Построения модели реального наведения ЗУР. Очевидно, что применение идеального наведения в реальности из определения невозможно. ЗУР в силу инерционности не способна обрабатывать требуемые положения мгновенно. В связи с этим необходима разработка системы наведения и стабилизации, функциональная схема которых представлена на рис. 14 [7]. Пропуская методику определения параметров блока формирования управляющего сигнала и параметров системы стабилизации, рассмотрим с помощью пакета Matlab.Simulink как будет вести себя ЗУР при использовании данных систем без подключения корректирующих фильтров (КФ). Изменения параметров ЗУР в виде графиков представлены на рис. 15–17.

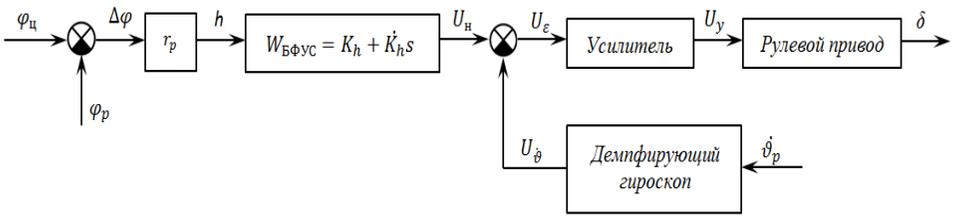


Рис. 14. Функциональная схема контура системы наведения и стабилизации без КФ

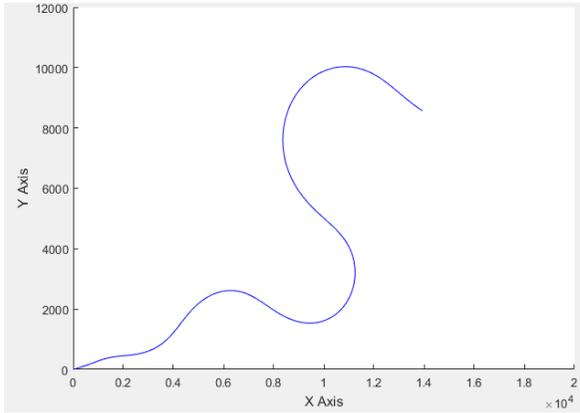
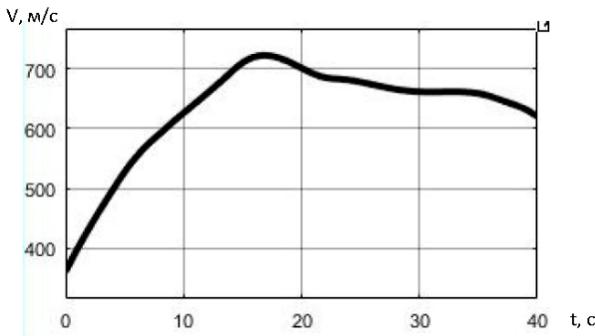
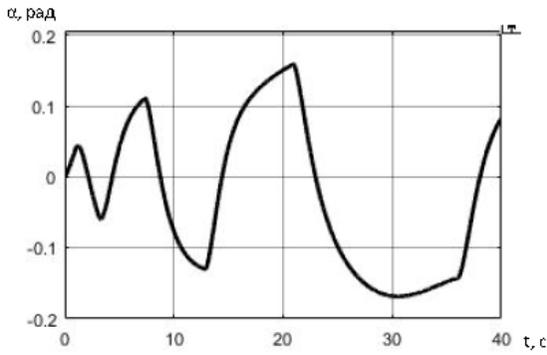


Рис. 15. Траектория ЗУР при реальном наведении без КФ



а



б

Рис. 16. Изменение скорости (а), угла атаки (б)

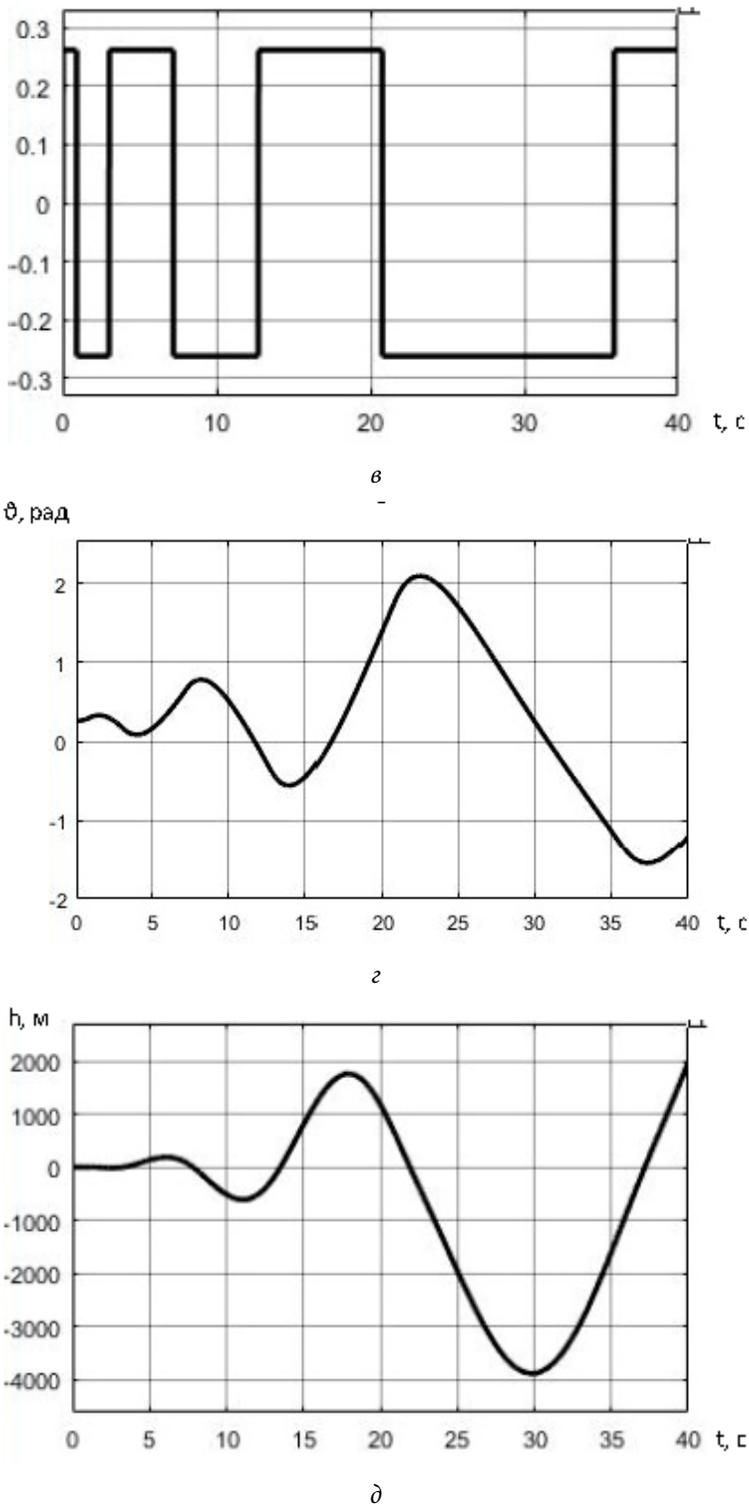
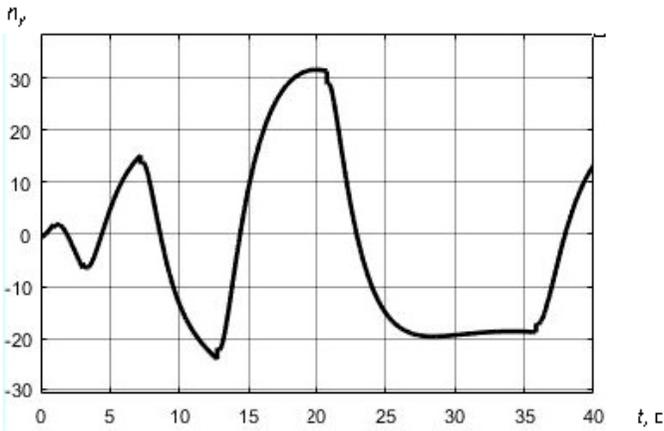


Рис. 16. Изменения угла отклонения рулей (θ), угла тангажа (z) и промах (δ) ЗУР при реальном наведении без КФ

Рис. 17. Перегрузка n_y

Как видно из рисунков, наведение ЗУР без использования КФ невозможно. Их разработка также может проводиться с помощью пакета Matlab.Simulink. Этому способствует то обстоятельство, что построение графиков переходного процесса, фазовых и частотных характеристик является встроенной функцией Matlab.Simulink. Примеры таких построений для передаточной функции неизменной части представлены на рис. 18 и 19.

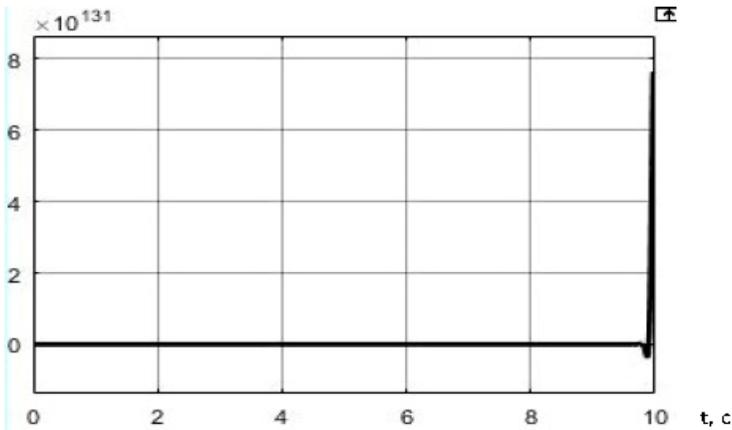


Рис. 18. Переходный процесс

Выводы. Использование пакета Matlab.Simulink при баллистических расчетах позволяет оперативно синтезировать модель требуемой системы с заданной точностью. Встроенные функции, такие как интегрирование, построение графиков, визуализация и другие специальные блоки значительно упрощают задачу и повышают функциональность синтезируемой модели. А блочно-модульное построение программы облегчает восприятие информации, упрощает процесс анализа и оптимизации их параметров [8]. В частности, при решении задачи проектирования КФ, основанных на интегро-дифференцирующих звеньях,

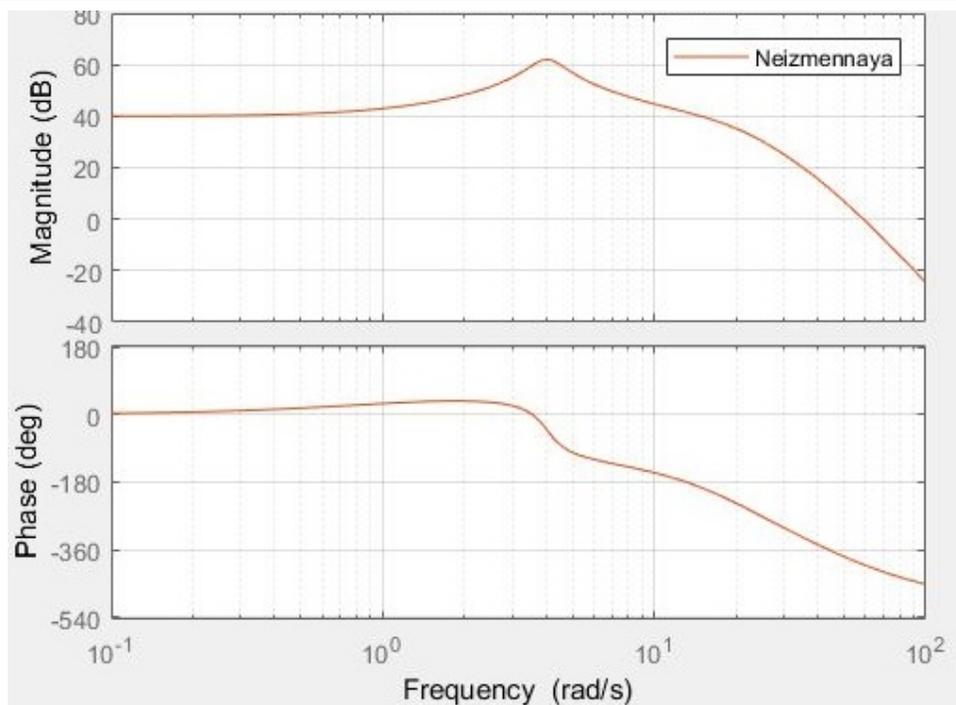


Рис. 19. Амплитудная (вверху) и фазовая (внизу) характеристики

использование данного пакета значительно облегчает работу, так как при этом не требуется использование сложных преобразований, которые необходимы при решении той же задачи на языке C++. Построение фазовых и частотных характеристик является встроенной функцией Matlab.Simulink, в то время как на других языках реализовать это невозможно либо очень трудоемко.

Литература

- [1] Веденичев И.В., Лысикова В.С. Исследование влияния алгоритма расчета матрицы направляющих косинусов на результаты численного моделирования движения ЛА. *Молодежный научно-технический вестник*, 2016, № 10. URL: <http://sntbul.bmstu.ru/doc/850294.html>.
- [2] Герман-Галкин С.Г. *MATLAB & Simulink. Проектирование мехатронных систем на ПК*. Санкт-Петербург, КОРОНА-Век, 2008, 368 с.
- [3] Илюхин С.Н., Москаленко В.О., Хлупнов А.И. Экспериментальные исследования влияния формы головной части пули на характеристики рассеивания при дозвуковых скоростях полета. *Аэрокосмический научный журнал*, 2015, № 5, с. 38–48. URL: <http://aerospace.elpub.ru/jour/article/view/27>.
- [4] Емельянова Н.С. Программное обеспечение экспериментальных исследований на баллистической трассе. *Молодежный научно-технический вестник*, 2014, № 6. URL: <http://sntbul.bmstu.ru/doc/723443.html>.
- [5] Алексеев Е.Р., Чеснокова О.В. *Решение задач вычислительной математики в пакетах Mathcad 12, MATLAB 7, Maple 9*. Москва, НТ Пресс, 2006, 496 с.

- [6] Илюхин С.Н., Беневольский С.В., Грабин В.В. *Формирование облика зенитной управляемой ракеты и динамический анализ её системы управления*. Москва, Изд-во МГТУ им. Н.Э. Баумана, 2012, 80 с.
- [7] Илюхин С.Н. Синтез системы наведения и контура стабилизации методом ЛАХ на при-мере произвольной модели ЗУР. *Молодежный научно-технический вестник*, 2012, № 7. URL: <http://sntbul.bmstu.ru/doc/467279.html>.
- [8] Афонин В.В. *Моделирование систем*. Москва, БИНОМ. Лаборатория знаний, 2010, 231 с.

Никанорова Маргарита Дмитриевна — студентка кафедры «Динамика и управление полетом ракет и космических аппаратов», МГТУ им. Н.Э. Баумана, Москва, Российская Федерация.

Веденичев Игорь Васильевич — студент кафедры «Динамика и управление полетом ракет и космических аппаратов», МГТУ им. Н.Э. Баумана, Москва, Российская Федерация.

Научный руководитель — С.Н. Илюхин, ассистент кафедры «Динамика и управление полетом ракет и космических аппаратов», МГТУ им. Н.Э. Баумана, Москва, Российская Федерация.

USING THE MATLAB.SIMULINK SOFTWARE SUITE FOR BALLISTIC ANALYSIS

M.D. Nikanorova

rnikanorova@gmail.com

I.V. Vedenichev

Vedenicheviv@gmail.com

Bauman Moscow State Technical University, Moscow, Russian Federation

Abstract

In this work we have simulated the flight of the material point in the MATLAB.Simulink environment coded in C++ and analyzed the results obtained. The benefits of applying this software suite have been revealed. We simulated the flight of the ideally homing surface-to-air missile and the surface-to-air missile real flight. The article proves the impossibility of using the guidance system without the correcting filters.

Keywords

MATLAB.Simulink environment, ballistics, numerical simulation, correcting filters

© Bauman Moscow State Technical University, 2017

References

- [1] Vedenichev I.V., Lysikova V.S. Research on effect of director cone matrix calculation algorithm on numerical simulation results of aircraft movement. *Molodezhnyy nauchno-tekhnicheskij vestnik*, 2016, no. 10. Available at: <http://sntbul.bmstu.ru/doc/850294.html>.
- [2] German-Galkin S.G. MATLAB & Simulink. Proektirovanie mekhatronnykh sistem na PK [MATLAB & Simulink. Computer programming of mechatronics systems]. Sankt-Petersburg, KORONA-Vek publ., 2008, 368 p.
- [3] Ilyukhin S.N., Moskalenko V.O., Khlupnov A.I. Experimentally studied influence of the bullet head shape on dispersion characteristics at subsonic airspeeds. *Aerokosmicheskij nauchnyy zhurnal* [Aerospace scientific journal], 2015, no. 5, pp. 38–48. Available at: <http://aerospace.elpub.ru/jour/article/view/27>.
- [4] Emel'yanova N.S. Software for experimental research on ballistic track. *Molodezhnyy nauchno-tekhnicheskij vestnik*, 2014, no. 6. Available at: <http://sntbul.bmstu.ru/doc/723443.html>.
- [5] Alekseev E.R., Chesnokova O.V. Reshenie zadach vychislitel'noy matematiki v paketakh Mathcad 12, MATLAB 7, Maple 9 [Solving problem of computational mathematics in Mathcad 12, MATLAB 7, Maple 9 software packages]. Moscow, NT Press publ., 2006, 496 p.
- [6] Ilyukhin S.N., Benevol'skiy S.V., Grabin V.V. Formirovanie oblika zenitnoy upravlyаемой ракеты i dinamicheskij analiz ee sistemy upravleniya [Forming conception of guided air defense rocket and dynamic analysis of its control system]. Moscow, Bauman Press, 2012, 80 p.
- [7] Ilyukhin S.N. Synthesis of pointing system and stabilization loop using log-magnitude and phase diagram method through the example of guided air defense rocket arbitrary model. *Molodezhnyy nauchno-tekhnicheskij vestnik*, 2012, no. 7. Available at: <http://sntbul.bmstu.ru/doc/467279.html>.
- [8] Afonin V.V. Modelirovanie system [System modelling]. Moscow, BINOM. Laboratoriya znaniy publ., 2010, 231 p.

Nikanorova M.D. — student, Department of Dynamics and Flight Control of Rockets and Spacecraft, Bauman Moscow State Technical University, Moscow, Russian Federation.

Vedenichev I.V. — student, Department of Dynamics and Flight Control of Rockets and Spacecraft, Bauman Moscow State Technical University, Moscow, Russian Federation.

Scientific advisor — Ilyukhin S.N., Assistant Lecturer of Dynamics and Flight Control of Rockets and Spacecraft of Department, Bauman Moscow State Technical University, Moscow, Russian Federation.