



Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Московский государственный технический университет
имени Н.Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н.Э. Баумана)

ФАКУЛЬТЕТ _____ ПРИБОРОСТРОИТЕЛЬНЫЙ _____

КАФЕДРА _____

РАСЧЕТНО-ПОЯСНИТЕЛЬНАЯ ЗАПИСКА К КУРСОВОЙ РАБОТЕ

НА ТЕМУ:

Синтез алгоритмов наведения и стабилизации центра масс
возвращаемого аппарата

Студент	ПС2-101 (Группа)	_____ (Подпись, дата)	Д. А. Родин (И.О.Фамилия)
Руководитель КР		_____ (Подпись, дата)	_____ (И.О.Фамилия)
Консультант		_____ (Подпись, дата)	_____ (И.О.Фамилия)

2020 г.

Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Московский государственный технический университет имени Н.Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н.Э. Баумана)

УТВЕРЖДАЮ

Заведующий кафедрой _____
(Индекс)

(Подпись)

(И.О. Фамилия)

« » _____ 2020 г.

З А Д А Н И Е
на выполнение курсовой работы

по дисциплине Системы управления(спецглавы)

Студент группы ПС2-101

Родин Дмитрий Андреевич

(фамилия, имя, отчество)

Тема курсовой работы

Синтез алгоритмов наведения и стабилизации центра масс возвращаемого аппарата

Задание

1. Синтез желаемой траектории центра масс ВА на участке посадки
2. Синтез алгоритмов стабилизации ЦМ относительно эталонной траектории
3. Проведение математического моделирования
4. Визуализация результатов моделирования
5. Оформление материалов по КП

Дата выдачи задания « » _____ 20__ г.

График выполнения работы: 25% к 3 нед., 50% к 9 нед., 75% к 12 нед., 100% к 15 нед.

Руководитель КР

(Подпись, дата)

(И.О.Фамилия)

Студент

(Подпись, дата)

Д. А. Родин

(И.О.Фамилия)

Примечание: Задание оформляется в двух экземплярах: один выдается студенту, второй хранится на кафедре.

СОДЕРЖАНИЕ

СОКРАЩЕНИЯ, ПРИНЯТЫЕ В ТЕКСТЕ	3
ВВЕДЕНИЕ	4
ОСНОВНАЯ ЧАСТЬ	5
1 ТРЕБОВАНИЯ ТЕХНИЧЕСКОГО ЗАДАНИЯ НА РАЗРАБОТКУ СУ ПТДУ	5
1.1 Назначение ПТДУ	7
1.2 Общий вид ВА	8
1.3 Конструкция и принцип действия ПТДУ	10
1.4 Характеристики органов управления	12
1.5 Требования к СУ	14
2 СХЕМА ДЕЙСТВИЯ	15
3 АЛГОРИТМ ОПРЕДЕЛЕНИЯ МОМЕНТА ВКЛЮЧЕНИЯ ПТДУ .	16
4 СИСТЕМЫ КООРДИНАТ	18
4.1 Матрица перехода из ПСК в ССК	19
5 МАТЕМАТИЧЕСКАЯ МОДЕЛЬ ДВИЖЕНИЯ ЦМ ВА	21
5.1 Расчет управляющих сил	22
5.2 Расчет аэродинамических сил	23
6 ОБЩАЯ ПОСТАНОВКА ЗАДАЧИ СИНТЕЗА СУ ПТДУ	24
6.1 Постановка задачи управления ЦМ	25
7 АЛГОРИТМ НАВЕДЕНИЯ	26
7.1 Формирование вектора требуемой ориентации ВА	28
8 ЗАКОНЫ УПРАВЛЕНИЯ В КОНТУРЕ СТАБИЛИЗАЦИИ	29

9	ЗАКОНЫ УПРАВЛЕНИЯ В КОНТУРЕ СТАБИЛИЗАЦИИ ОБЪЕКТА	31
10	СИНТЕЗ АЛГОРИТМОВ СТАБИЛИЗАЦИИ	33
10.1	Метод модального управления	33
10.2	Постановка задачи слежения по ЦМ	36
11	МОДЕЛИРОВАНИЕ ПОСАДКИ ВА	37
	ЗАКЛЮЧЕНИЕ	40
	СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ	41
	ПРИЛОЖЕНИЕ	41

СОКРАЩЕНИЯ, ПРИНЯТЫЕ В ТЕКСТЕ

БСК	блок силовой коммутации
БУК	блок усиления и коммутации
БЦВМ	бортовая цифровая вычислительная машина
БЭ	блок электроники
ВА	возвращаемый аппарат
ДДУ	дискретный датчик угла
ИТ	источник тока
НС	наблюдатель состояния
ОУ	объект управления
ПТДУ	посадочная твердотопливная двигательная установка
ПТК НП	пилотируемый транспортный корабль нового поколения
ПУ	посадочное устройство
РДТТ	ракетный двигатель твердого топлива
РМ	рулевая машинка
РП	рулевой привод
РПУ	релейное пороговое устройство
РС	регулятор состояния
СУ	система управления
СУБ	сопловые управляемые блоки
ТП	телеметрический потенциометр
ТРТ	твердое ракетное топливо
УПК	устройство преобразования кода Грея в двоичный код
УСДК	устройство сравнения двоичных кодов
ЭД	электродвигатель
ЭП	электрохимический привод

ВВЕДЕНИЕ

Целью данного курсового проекта является синтез алгоритма наведения и стабилизации центра масс возвращаемого аппарата. Разработка алгоритмов и программного обеспечения математического моделирования замкнутой системы управления в MatLab. Проведение сравнительного анализа результатов моделирования.

Под синтезом системы автоматического управления понимается направленный расчет, имеющий конечной целью отыскание рациональной структуры системы и установление оптимальных величин параметров ее отдельных звеньев.

Синтез можно трактовать как инженерную задачу, сводящуюся к такому построению системы, при котором обеспечивается выполнение технических требований к ней. Подразумевается, что из многих возможных решений инженер, проектирующий систему, будет выбирать те, которые являются оптимальными с точки зрения существующих конкретных условий и требований к габаритам, весу, простоте, надежности и т. п.

При инженерном синтезе системы автоматического управления необходимо обеспечить, во-первых, требуемую точность и, во-вторых, приемлемый характер переходных процессов.

Обеспечение приемлемых переходных процессов оказывается почти всегда более трудным вследствие большого числа варьируемых параметров и многозначности решения задачи демпфирования системы. Поэтому существующие инженерные методы часто ограничиваются решением только второй задачи, так как обеспечение требуемой точности может быть достаточно просто сделано на основании использования существующих критериев точности и совершенствования их практически не требуется.

В настоящее время для целей синтеза систем автоматического управления широко используются вычислительные машины, позволяющие производить полное или частичное моделирование проектируемой системы

ОСНОВНАЯ ЧАСТЬ

1 Требования технического задания на разработку СУ ПТДУ

1. Рассматривается схема газосвязанной ПТДУ с регулированием тяги
2. Основные потребные параметры ПТДУ определяются значениями

- Диапазон суммарной тяги $R_{\Sigma} = 9.0 - 22.0$ тс, при этом предполагается возможность реализации управления тягой по заданным алгоритмам в зависимости от реализованных условий посадки, обусловленных разбросом параметров атмосферы, аэродинамических, массовых, инерционных, центровочных характеристик возвращаемого аппарата, параметров, траектории и др
- Суммарный импульс тяги по осям всех сопел $I_{\Sigma} = 260 \text{ тс} \cdot \text{с}$
- Максимальное время работы 30 с.

3. Рассматривается применение в составе ПТДУ от 8 до 12 сопловых управляющих блоков (СУБ), Каждый СУБ имеет два расходных сопла, одно из которых направлено вдоль продольной оси ВА, второе под углом (30... 45) град к поперечной плоскости ВА. Регулирование расхода через каждое сопло осуществляется по командам от систем управления дифференцированно с помощью регулятора механического типа, вал которого кинематически связан с рулевой машинкой по командам управления.
4. Сопла, направленные вдоль продольной оси ВА (сопла первой группы), задействуются на основном участке торможения от момента начала работы ПТДУ и до высоты 40 м, с которой начинается приземный участок торможения. Группа обеспечивает гашение линейной

скорости ВА на этом участке, а также управление относительно ЦМ по каналам рысканья и тангажа.

5. Сопла, направленные под углом (30...45) грд к поперечной плоскости ВА (сопла второй группы), введены в состав ПТДУ для минимизации воздействия струй работающей ПТДУ на грунт. Сопла второй группы задействуются на приземном участке, а также для управления по каналам рысканья и тангажа на основном участке (при необходимости) и на приземном.
6. Должна быть проработана целесообразность и, при необходимости, реализация управления по каналу крена средствами ПТДУ при ее работе.

1.1 Назначение ПТДУ

Посадочная твердотопливная двигательная установка (ПТДУ) предназначена для снижения скорости возвращаемого орбитального аппарата (ВА) на участке приземления от величин, соответствующих установившейся скорости спуска, до заданных значений к моменту касания земли. При этом должна обеспечиваться возможность реализации управления тягой по заданным алгоритмам в зависимости от реализованных условий посадки по командам системы управления ВА.

Под участком приземления понимается участок спуска ВА, начинающийся в момент включения ПТДУ и заканчивающийся в момент касания первой опоры посадочного устройства (ПУ) ВА с поверхностью полигона. Приведение ВА на полигон обеспечивается управлением ВА до включения ПТДУ. Под полигоном понимается подготовленная посадочная площадка диаметром несколько км. Включение ПТДУ должно производиться на высоте 1 км над поверхностью полигона.

1.2 Общий вид ВА

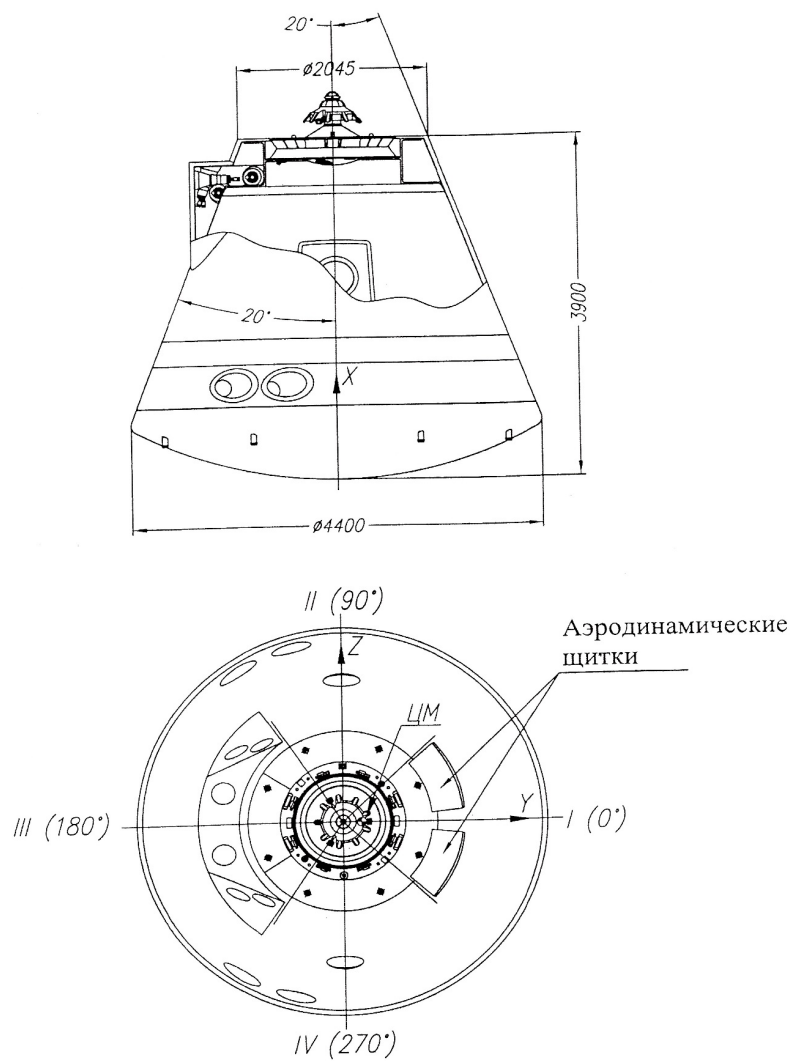


Рис. 1: Общий вид ВА

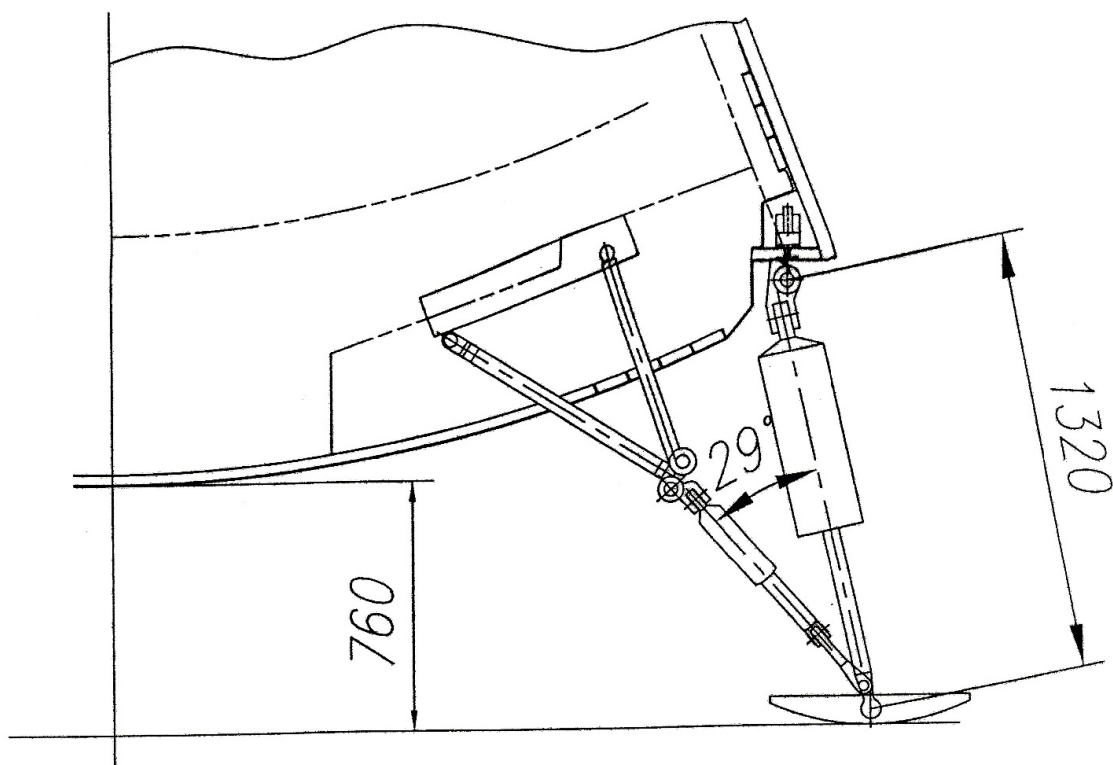


Рис. 2: Посадочная конфигурация ВА с выпущенным посадочным устройством

1.3 Конструкция и принцип действия ПТДУ

ПТДУ представляет собой РДТТ с регулируемой тягой по величине и направлению. На рисунке показана схема ПТДУ. Он состоит из:

- двух корпусов типа ”кокон” с наполнителями ТРТ ”Г-З”
- 16 односопловых управляющих блоков (СУБ)
- системы газоходов, газосвязывающего корпуса с наполнителями и СУБ
- двух пусковых двигателей
- блока датчиков давления системы измерения давления
- рулевого привода (электромеханического или газогидравлического)

Сопла всех СУБ снабжены собственными регуляторами расхода, управляемыми собственными рулевыми машинками. Расположение и маркировка органов управления соответствует рисунку 6.1.

ПТДУ многорежимна. Переход с режима на режим, а также стабилизация давления на любом режиме, обеспечивается изменением газопритока от поверхности горения за счет изменения скорости горения наполнителя путем кратковременного изменения суммарной площади минимальных сечений сопел.

Создание управляющих сил по каналам тангажа и рыскания на каждом режиме обеспечивается за счет перераспределения расхода продуктов сгорания ТРТ между соплами путем изменения площади минимального сечения каждого сопла при сохранении постоянной суммарной площади минимальных сечений всех сопел:

$$\mu F_{\Sigma} = \sum_{j=1}^{16} \mu F_j \approx const \quad (1)$$

где μF_{Σ} - требуемая суммарная эффективная площадь минимальных сечений сопел на соответствующем режиме работы ПТДУ.

μF_j - текущая эффективная площадь минимального сечения j -го сопла.

При необходимости должна быть обеспечена возможность отсечки тяги ПТДУ по команде СУ. При этом для прекращения горения наполнителей должно выполняться условие:

$$\mu F_{\Sigma} = \sum_{j=1}^{16} \mu F_j^{max} \quad (2)$$

где μF_j^{max} - максимальная эффективная площадь минимального сечения j - го сопла.

В случае отказа одного из СУБ или отказа ПМ площадь минимального сечения этого СУБ при дальнейшей работе ПТДУ должна оставаться постоянной, при это должна выполняться заданная циклограмма суммарной тяги R_{Σ} и управление остальными СУБ в заданном диапазоне R .

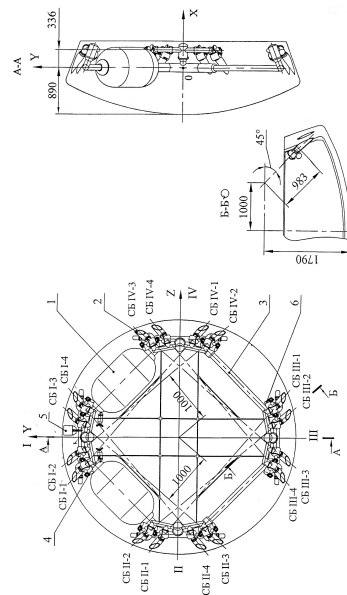


Рис. 3: Схема ПТДУ

1.4 Характеристики органов управления

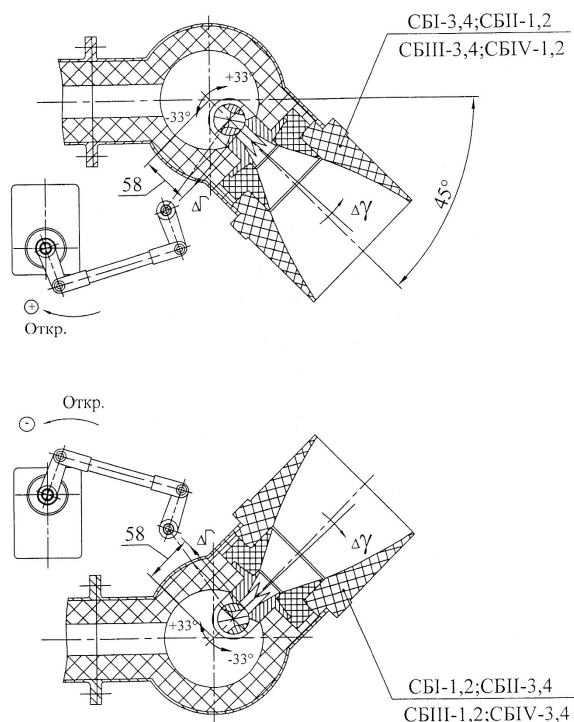


Рис. 4: Кинематическая схема СУБ с РП

Зависимость эффективной площади минимального сечения единичного сопла СУБ (μF_j) от угла поворота вала РМ (δ_j) представлена на рисунке (5)

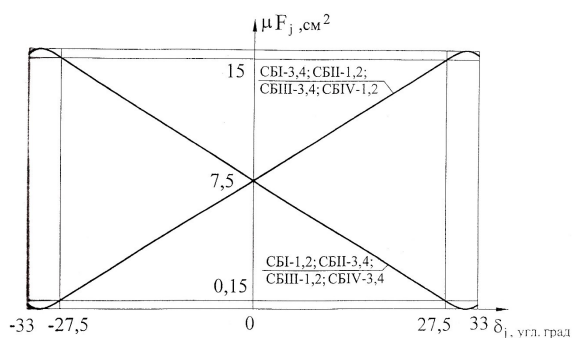


Рис. 5: Зависимость эффективной площади сечения единичного сопла (μF_j) от угла поворота вала РМ (δ_j)

При работе ПТДУ используется только линейный участок зависимости $\mu F_j(\delta_j)$.

На участке от -27.5° до 27.5° для данной расходной характеристики допускается линейная аппроксимация:

$$\delta_j = 3.667 \cdot \mu F_j - 27.5 \quad (3)$$

Систематическое линейное и угловое отклонение вектора тяги в единичном сопле в зависимости от угла поворота РМ от геометрической оси сопла за счет прямоугольной формы площади минимального сечения регулятора сопла приведено на (5).

Отклонение происходит в плоскости, проходящей через геометрическую ось сопла перпендикулярно продольной оси корпуса СУБ.

Случайное отклонение вектора тяги от номинального положения геометрической оси сопла в плоскости минимального сечения составляет: линейное 1.5 мм, угловое 1.5° .

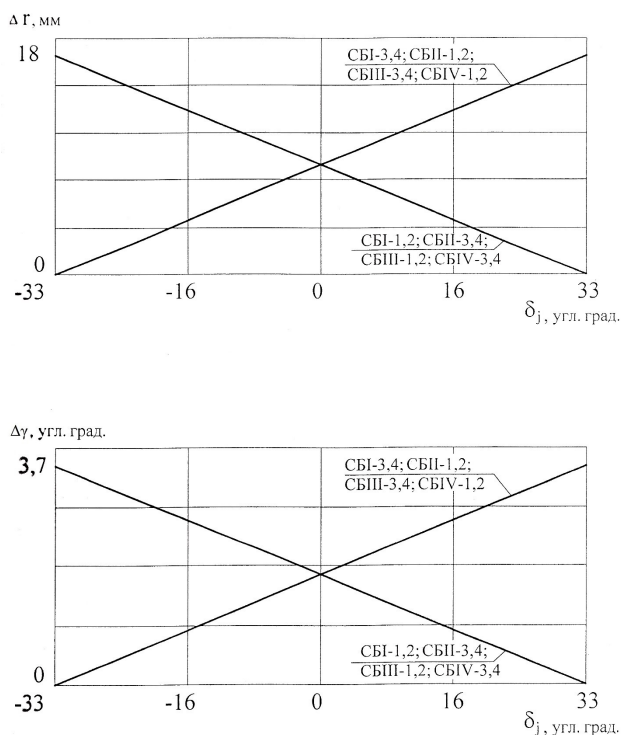


Рис. 6: Линейное (Δr) и угловое ($\Delta \gamma$) отклонение линии действия вектора тяги от геометрической оси сопла в зависимости от угла поворота вала РМ

1.5 Требования к СУ

Исходными данными на разработку технических предложений по алгоритмам СУ, СУ ПТДУ для управления на участке посадки возвращаемого аппарата определены следующие положения:

- Начальные условия при запуске ПТДУ:
 - угол наклона траектории $-70^\circ \div -90^\circ$
 - скорость $80 \div 120 \frac{\text{м}}{\text{с}}$
 - угол атаки $8^\circ \div 20^\circ$
 - угловые скорости до ± 50 угл. град/с по трем углам одновременно
 - высота $700 \div 1200$ м.
- Условия окончания:
 - на высоте 20 м начинается участок вертикального движения, который завершается касанием. Скорость во время касания $2 \div 3 \frac{\text{м}}{\text{с}}$
- суммарный импульс тяги:
$$\int_0^T R(r) dt = 280 \text{ Тс} \cdot \text{с}$$
- Ограничения на СУ в вертикальном канале ВА:
 - отклонение от требуемой траектории ± 5 м

2 Схема действия

Перед запуском ПТДУ проводится тестирование тракта исполнительных механизмов (РМ), производится установка стартовых величин эффективных сечений сопловых блоков (СБ). По завершению тестового режима подается сигнал на зажигание ПТДУ. В командных сигналах на РМ СБ учитываются демпфирующие составляющие закона управления угловым положением аппарата. Таким образом, уже при запуске ДУ предотвращается развитие неустойчивого процесса. Это участок успокоения - т. е. гашение угловых скоростей и разворот аппарата продольной осью вдоль вектора требуемой ориентации, вырабатываемого терминальным регулятором. По крену аппарат разворачивается таким образом, что вектор \bar{Z}_1 аппарата ориентируется ортогонально вектору $\bar{\xi}$ ПСК в положительном направлении $\bar{\zeta}$. Затем по оперативно рассчитываемой программе аппарат затормаживается и выводится в точку посадки на высоту 20м с вертикальной скоростью $-2 \div -3 \frac{\text{м}}{\text{с}}$. Управление на этом участке заключается в поддержании постоянной вертикальной скорости и стабилизации аппарата в горизонтальной плоскости.

3 Алгоритм определения момента включения ПТДУ

Исходя из независимости движений в ортогональных направлениях и приоритете вертикального канала в процессе управления, выбор момента включения ДУ определяется высотой, скоростью движения аппарата и необходимостью обеспечения наиболее предпочтительного режима работы ПТДУ.

Предпочтительным режимом является работа в середине линейного диапазона регулирования тяги, так как при действии возмущений такой режим создает наилучшие условия для управления.

В связи с тем, что аппарат на момент включения ПТДУ движется практически с установившейся скоростью, суммарная потенциальная и кинетическая энергия, которую необходимо рассеять с помощью ПТДУ, убывает при снижении аппарата:

$$E(h) = \frac{mv^2}{2} + mgh \quad (4)$$

где $v \approx const$

Принимая равенство совершаемой двигателем работы и энергии аппарата, погашаемой на участке спуска при величине тяги двигателя, соответствующей середине диапазона регулирования $E = R_{cp}\eta_0$, получаем высоту включения ПТДУ:

$$\eta_0 = \frac{mv^2}{2} \cdot \frac{1}{R_{cp} + C_x \rho S_m \frac{v_\eta^2}{6} - mg} \quad (5)$$

здесь $R_{cp} = 12\text{Тс}$.

Этот функционал может вычисляться на борту и при возможном разбросе начальной скорости v_{η_0} в пределах $80 \div 120 \frac{\text{м}}{\text{с}}$, высота включения ПТДУ лежит в пределах $\eta_0 = 400 \div 800\text{м}$. Предполагая, что в вертикальном канале реализуется движение с постоянным замедлением, т. е. $\dot{V}_\eta = const$, получаем интегралы движения:

$$\eta = \eta_0 - V_\eta \cdot t + \frac{\dot{V}_\eta \cdot t^2}{2} \quad (6)$$

$$V_{\eta} = -V_{\eta_0} + \dot{V}_{\eta} \cdot t \quad (7)$$

откуда

$$\dot{V}_{\eta} = \frac{V_{\eta_0}^2}{2\eta_0} \quad (8)$$

$$T = -\frac{V_{\eta_0}}{\dot{V}_{\eta}} \quad (9)$$

Среднее значение вертикального ускорения, обеспечивающего затормаживание на заданной высоте $\dot{v}_{\eta} \approx 12 \div 20 \frac{\text{м}}{\text{с}^2}$, и время торможения $T \approx 8 - 10 \text{с}$. При этом минимальный суммарный импульс тяги, расходуемый на затормаживание аппарата равен $\approx 150 \text{Тс} \cdot \text{с}$. На участке вертикального спуска расходуется еще $\approx 70 \div 80 \text{Тс} \cdot \text{с}$.

Минимальная высота включения ПТДУ, при которой еще возможно погасить энергию аппарата $h_0 \approx 400 \text{м}$, время торможения $\approx 7 \text{с}$, тяга $R_{\text{max}} = 22 \text{Тс}$. При этом перемещения в горизонтальной плоскости исключаются. Весь импульс при этом сбрасывается в вертикальном канале управления.

4 Системы координат

В работе используются посадочная система координат и связанная с ВА система координат.

Задачи наведения решаются в посадочной системе координат (ПСК):

- начало координат - в точке посадки;
- Ось $O\eta$ - по местной вертикали;
- Ось $O\xi$ - по касательной к меридиану в точке посадки и направлена на север;
- Ось $O\zeta$ - дополняет систему до правой;

Силы и моменты, действующие на ЛА рассмотрим в связанной системе координат $OX_1Y_1Z_1$ - правая прямоугольная система координат, связанная с изделием (ССК):

- Ось OX_1 - совпадает с продольной осью симметрии изделия и направлена в сторону хвостовой части изделия;
- Ось OY_1 - направлена в сторону I строительной плоскости изделия;
- Ось OZ_1 - дополняет систему координат до правой;

4.1 Матрица перехода из ПСК в ССК

Для составления матрицы перехода из ПСК в ССК необходимо составить элементарные матрицы последовательного поворота на один угол и перемножить их. Повороты будем осуществлять против часовой стрелки, т. е. в положительном направлении, в следующей последовательности: $\vartheta \rightarrow \psi \rightarrow \varphi$ как показано на рис.

Для первого поворота на угол ϑ , получим:

$$A_{\vartheta} = \begin{bmatrix} \cos \vartheta & \sin \vartheta & 0 \\ -\sin \vartheta & \cos \vartheta & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad (10)$$

для второго поворота на угол ψ

$$A_{\psi} = \begin{bmatrix} \cos \psi & 0 & -\sin \psi \\ 0 & 1 & 0 \\ -\sin \psi & 0 & \cos \psi \end{bmatrix}, \quad (11)$$

для третьего поворота на угол ϕ

$$A_{\varphi} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \varphi & \sin \varphi \\ 0 & -\sin \varphi & \varphi \end{bmatrix} \quad (12)$$

Произведение трех матриц имеет вид:

$$A = A_{\varphi} A_{\psi} A_{\vartheta} \quad (13)$$

В результате перемножения получим матрицу перехода от ПСК к ССК:

$$A = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix}, \quad (14)$$

где

$$\begin{aligned}a_{11} &= \cos \vartheta \cdot \cos \varphi \\a_{12} &= \cos \vartheta \cdot \sin \varphi \cdot \sin \psi - \sin \vartheta \cdot \cos \varphi \\a_{13} &= \cos \vartheta \cdot \sin \psi \cdot \cos \varphi + \sin \vartheta \cdot \sin \varphi \\a_{21} &= \sin \vartheta \cdot \cos \psi \\a_{22} &= \sin \vartheta \cdot \sin \varphi \cdot \sin \psi + \cos \vartheta \cdot \cos \varphi \\a_{23} &= \sin \vartheta \cdot \sin \psi \cdot \cos \varphi - \cos \vartheta \cdot \sin \varphi \\a_{31} &= -\sin \psi \\a_{32} &= \cos \psi \cdot \sin \varphi \\a_{33} &= \cos \psi \cdot \cos \varphi\end{aligned}\tag{15}$$

Формула перехода имеет вид:

$$\begin{bmatrix} \xi \\ \eta \\ \zeta \end{bmatrix} = A \cdot \begin{bmatrix} X_1 \\ Y_1 \\ Z_1 \end{bmatrix}\tag{16}$$

5 Математическая модель движения ЦМ ВА

Модель движения ЦМ запишем в следующем виде:

$$\begin{cases} \dot{W}_{X1} = \frac{F_{X1}}{m} \\ \dot{W}_{Y1} = \frac{F_{Y1}}{m} \\ \dot{W}_{Z1} = \frac{F_{Z1}}{m} \end{cases}, \quad (17)$$

где \dot{W}_{X1} , \dot{W}_{Y1} , \dot{W}_{Z1} - проекции кажущегося ускорения на оси связанной системы координат, F_{X1} , F_{Y1} , F_{Z1} - проекции сил действующих на ВА, m - масса ВА.

$$\begin{aligned} F_{X1} &= X - R_{\Sigma}, \\ F_{Y1} &= F_{\text{упр}Y1} + Y, \\ F_{Z1} &= F_{\text{упр}Z1} + Z, \end{aligned} \quad (18)$$

где R_{Σ} - суммарная сила тяги действующая по оси OX_1 связанной системы координат, $F_{\text{упр}Y1}$ - управляющая сила действующая по оси OY_1 связанной системы координат, $F_{\text{упр}Z1}$ - управляющая сила действующая по оси OZ_1 связанной системы координат, X - продольная составляющая аэродинамической силы, Y - нормальная составляющая аэродинамической силы, Z - поперечная составляющая аэродинамической силы.

5.1 Расчет управляющих сил

На ЛА во время полета также действуют управляющие силы и моменты, которые создаются с помощью СУБ

Угол наклона сопел равен $\alpha_c = 5.73^\circ$ показан на рисунке , максимальная суммарная тяга равна

$$R_{\Sigma_{max}} = 22 \text{ Тс}$$

Максимальная тяга i - го сопла

$$R_{\max i} = 2.75 \text{ Тс}$$

Минимальная суммарная тяга

$$R_{\Sigma_{\min}} = 8 \text{ Тс}$$

Минимальная тяга i - го сопла

$$R_{\min i} = 1 \text{ Тс}$$

5.2 Расчет аэродинамических сил

При полете ЛА в атмосфере на них действует сопротивление воздуха, называемое аэродинамическим.

Аэродинамическая сила R_A складывается из сил давления воздуха, направленных по нормальям к поверхности ЛА, и сил трения воздуха касательных к ней.

Для определения R_A используют формулу:

$$R_A = C_R \cdot q \cdot S, \quad (19)$$

где $q = \frac{\rho \cdot V^2}{2}$ - скоростной напор набегающего невозмущенного потока, V - скорость набегающего потока, ρ - плотность атмосферы, S_M - характерная площадь ракеты, M - число маха ракеты, C_R - безразмерный аэродинамический коэффициент, зависящий от формы ракеты.

Запишем формулу для нахождения аэродинамической силы в плоскости OY_1Z_1 :

$$R_{Y_1Z_1} = C_{Y_1}(\alpha^*, M) \cdot q \cdot S_M \quad (20)$$

Продольную составляющую аэродинамической силы запишем следующим образом:

$$X = C_{X_1} \cdot q \cdot S_M \quad (21)$$

Нормальная составляющая аэродинамической силы:

$$Y = R_{Y_1Z_1} \cdot \cos \varphi \quad (22)$$

Поперечная составляющая аэродинамической силы:

$$Z = -R_{Y_1Z_1} \sin \varphi \quad (23)$$

6 Общая постановка задачи синтеза СУ ПТДУ

Исходя из требований ТЗ, при н. у. которые были определены ранее решенной задачей определения момента включения ПТДУ, требуется переместить объект из одной из возможных точек области н. у. в заданную точку пространства (точка зависания) за заданное время. В связи с тем, что органами управления являются не поворачиваемые сопла, т. о. сама задача определяется движением ЦМ и угловым движением. В связи с компоновкой сопловых блоков ВА эти задачи взаимосвязаны. Для того чтобы упростить задачу на данном этапе синтеза будем считать что есть приведенные моменты и приведенные силы. Распределения сил и моментов по соплам это задача следующего этапа. Чтобы синтез управления упростить мы декомпозируем систему на две независимые подсистемы: управления ЦМ и Управления угловым положением на основе разделения движений. Считаем, что динамика требуемой угловой ориентации на порядок быстрее чем движения ЦМ.

При этом каждой из этих подсистем управления выдвигаются свои требования по качеству:

1. Время переходного процесса
2. Отсутствие перерегулирования
3. Точность вывода в точку при вариации н. у. в области, которая ранее была определена алгоритмом включения ПТДУ

6.1 Постановка задачи управления ЦМ

Цель управления:

$$\|x(t) - x_*(t)\| \leq \Delta_x, \forall t \geq \hat{t} \quad (24)$$

где $x(t) = (\xi \dot{\xi} \eta \dot{\eta} \zeta \dot{\zeta})^T$ - вектор состояния объекта управления относительно центра масс, $x_*(t)$ - желаемая траектория спускаемого аппарата с заданными показателями качества.

Это была общая постановка задачи управления. Для того чтобы решить данную задачу мне нужно прежде всего сформировать желаемую траекторию и стабилизироваться относительно её.

Таким образом, задача распадается на две подзадачи:

1. Синтез желаемой кинематической траектории с учетом желаемой динамики движения по данной траектории.
2. Стабилизация ЦМ ВА относительно желаемой траектории (задача слежения)

7 Алгоритм наведения

Мы имеем начальные условия и конечные координаты траектории на момент времени T . Кинематическая функция должна быть гладкой функцией хотя бы второго порядка.

Будем искать решение кинематической траектории в виде многочлена на единице меньше, чем количество краевых условий. Это связано с тем, что у многочлена степени n в общем случае имеется $(n + 1)$ параметров, которые подлежат выбору.

Краевыми условиями являются начальные и конечные координаты, скорости и ускорения. Т.о. в качестве функций описывающих траектории движения выберем полиномы пятого порядка:

$$\alpha(t) = \alpha_0 + \alpha_1 t + \alpha_2 t^2 + \alpha_3 t^3 + \alpha_4 t^4 + \alpha_5 t^5 \quad (25)$$

$$\dot{\alpha}(t) = \alpha_1 + \alpha_2 t + \alpha_3 t^2 + \alpha_4 t^3 + \alpha_5 t^4 \quad (26)$$

$$\ddot{\alpha}(t) = 2\alpha_2 + 6\alpha_3 t + 12\alpha_4 t^2 + 20\alpha_5 t^3 \quad (27)$$

Где $\alpha(t)$ любая из координат $\xi(t)$, $\eta(t)$, $\zeta(t)$, t - время, на котором действуют коэффициенты $a_0 - a_5$.

Задаваясь начальными и конечными значениями координат $\alpha(0)$, $\dot{\alpha}(0)$, $\ddot{\alpha}(0)$, и $\alpha(T)$, $\dot{\alpha}(T)$, $\ddot{\alpha}(T)$ получаем три тройки уравнений относительно $a_{3\xi}$, $a_{3\eta}$, $a_{3\zeta}$, $a_{4\xi}$, $a_{4\eta}$, $a_{4\zeta}$, $a_{5\xi}$, $a_{5\eta}$, $a_{5\zeta}$ добавляя к ним коэффициенты $a_{0\xi}$, $a_{0\eta}$, $a_{0\zeta}$, $a_{1\xi}$, $a_{1\eta}$, $a_{1\zeta}$, $a_{2\xi}$, $a_{2\eta}$, $a_{2\zeta}$, которые определяются начальными условиями,

имеем следующее решение для определения многочленов пятого порядка:

$$\begin{aligned}
 a_0 &= \alpha(0), \\
 a_1 &= \dot{\alpha}(0) \\
 a_2 &= \ddot{\alpha}(0) \\
 a_3 &= \frac{10 \cdot (a(T) - a(0))}{T^3} - \frac{4 \cdot \dot{a}(T) + 6 \cdot \dot{a}(0)}{T^2} + \frac{\ddot{a}(T) - 3 \cdot \ddot{a}(0)}{2 \cdot T} \quad (28) \\
 a_4 &= \frac{-15 \cdot (a(T) - a(0))}{T^4} + \frac{7 \cdot \dot{a}(T) - 8 \cdot \dot{a}(0)}{T^3} + \frac{1.5 \cdot \ddot{a}(T) - \ddot{a}(0)}{T^3} \\
 a_5 &= \frac{6 \cdot (a(T) - a(0))}{T^5} + \frac{3 \cdot \dot{a}(T) - \dot{a}(0)}{T^4} + \frac{\ddot{a}(T) - \ddot{a}(0)}{T^3}
 \end{aligned}$$

T - время, оставшееся до окончания процесса, начальное значение которого определяется в момента запуска ПТДУ.

7.1 Формирование вектора требуемой ориентации ВА

Требуемая ориентация вектора тяги находится из компонент вектора требуемого кажущегося ускорения в виде соответствующих направляющих косинусов

$$\bar{e}^R = (e_\xi^R, e_\eta^R, e_\zeta^R) \quad (29)$$

$$\begin{aligned} \dot{W}_\xi^{\text{TP}} &= \dot{V}_\xi, \\ \dot{W}_\eta^{\text{TP}} &= \dot{V}_\eta + g, \\ \dot{W}_\zeta^{\text{TP}} &= \dot{V}_\zeta, \end{aligned} \quad (30)$$

Рассчитываем требуемое кажущиеся ускорение ЛА

$$\dot{W}_{\text{TP}} = \sqrt{\dot{W}_\xi^{\text{TP}^2} + \dot{W}_\eta^{\text{TP}^2} + \dot{W}_\zeta^{\text{TP}^2}} \quad (31)$$

Так как требуемое направление оси X противоположно направлению тяги, выражения принимают следующий вид:

$$\begin{aligned} e_\xi^R &= -\frac{\dot{W}_\xi^{\text{TP}}}{\dot{W}_{\text{TP}}}, \\ e_\eta^R &= -\frac{\dot{W}_\eta^{\text{TP}}}{\dot{W}_{\text{TP}}}, \\ e_\zeta^R &= -\frac{\dot{W}_\zeta^{\text{TP}}}{\dot{W}_{\text{TP}}} \end{aligned} \quad (32)$$

8 Законы управления в контуре стабилизации

Для управления двигателем вычисляется командный сигнал, состоящий из четырех составляющих

$$\delta_R = \delta^0 + K_0 \cdot (W_c(j\lambda) \cdot P_{cp} - \bar{P}_{пр}(\bar{R}_{тр})) + K_1 \cdot (\dot{W}_{X_1} - \dot{W}_{тр}) + K_2 \cdot (V_\eta - V_{\eta_{тр}}) \quad (33)$$

P_{cp} - величина давления в камере сгорания. Увеличению командного сигнала соответствует открытие СБ.

Первая составляющая δ^0 - уставка, определяющая режим работы ПТДУ, задается априорно и интерполируется в функции требуемой величины тяги по ходу процесса.

Вторая составляющая контролирует процесс в камере сгорания; $W_c(j\lambda)$ - интегро-дифференцирующий цифровой фильтр, стабилизирующий процесс в КС.

Третья составляющая следит за выполнением основной задачи - созданием требуемого кажущегося ускорения по продольной оси изделия.

Четвертая составляющая используется на финишном участке при реализации постоянной вертикальной скорости.

Интегральная составляющая в законе управления отсутствует ввиду переменного характера требуемой величины тяги. Требуемая точность регулирования достигается достаточным коэффициентом усиления.

Командный сигнал уставки ограничивается по величине отрицательного градиента

$$\begin{aligned} \bar{P}_{пр}(nT_c) &= \bar{P}_{пр}[(n-1)T_c] + \Delta P_{пр} \\ \Delta P_{пр} &= P_{пр}(R_{тр}) - \bar{P}_{пр}(R_{тр}) \end{aligned} \quad (34)$$

если $\dot{P}_{пр} < 0$ и $|\Delta P| > \Delta P(P_{пр})$

$$\Delta P = -\Delta P(P_{пр}) \quad (35)$$

если $\dot{P}_{пр} \geq 0$, то $\Delta P \Rightarrow \Delta P^{\max}$

$$\dot{P}_{пр} = P_{пр}(nT_c) - P_{пр}[(n-1)T_c] \quad (36)$$

$\Delta P(P_{\text{пр}})$ - интерполируется как монотонно возрастающая функция при возрастании $P_{\text{пр}}$; T_c - цикл решения задачи стабилизации давления, $T_c = 0.25 \cdot T_0$; T_{0c} - цикл выдачи сигналов ССД на органы управления, $T_{0c} = 0.5 \cdot T_0$.

На спад давления обеспечивается ограничение градиента спада тяги на уровне 3000

Командный сигнал δ_R ограничивается с учетом допустимого отклонения органов управления

$$\bar{\delta})_R = \begin{cases} \delta_R, & \text{если } |\delta_R| < \delta_{\text{ОГР}} \\ \delta_{\text{ОГР}} \cdot \text{sign} \delta_R, & \text{если } |\delta_R| \geq \delta_{\text{ОГР}} \end{cases} \quad (37)$$

9 Законы управления в контуре стабилизации объекта

Закон управления угловым поворотом продольной оси изделия подкрепляется для увеличения точности составляющими нормальной и боковой стабилизации

$$\begin{aligned}\delta_T &= \left(K_{\vartheta}(P) \cdot \Delta\vartheta + K_{\vartheta}(P) \cdot \tau \cdot \omega_z + K_{\dot{W}}(P) \cdot \Delta\dot{W}_{Y_1} \right) \cdot W_{\vartheta}(j\lambda_0) \\ \delta_P &= \left(K_{\psi}(P) \cdot \Delta\psi + K_{\psi}(P) \cdot \tau \cdot \omega_y - K_{\dot{W}}(P) \cdot \Delta\dot{W}_{Z_1} \right) \cdot W_{\psi}(j\lambda_0)\end{aligned}\quad (38)$$

Составляющей нормальной и боковой стабилизации контролируют движение объекта в горизонте:

$$\begin{aligned}\Delta\dot{W}_{Y_1} &= \dot{W}_{Y_1} - \dot{W}_{Y_1}^{TP} \\ \Delta\dot{W}_{Z_1} &= \dot{W}_{Z_1} - \dot{W}_{Z_1}^{TP} \\ \dot{W}_{Y_1}^{TP} &= C_{12} \cdot \dot{W}_{\xi}^{TP} + C_{32} \cdot \dot{W}_{\zeta}^{TP} \\ \dot{W}_{Z_1}^{TP} &= C_{13} \cdot \dot{W}_{\xi}^{TP} + C_{33} \cdot \dot{W}_{\zeta}^{TP}\end{aligned}\quad (39)$$

В канале крена закон управления

$$\delta_{\varphi} = (K_{\varphi}(P) \cdot \varphi + K_{\varphi}(P) \cdot \tau_{\varphi} \cdot \omega_{x_1}) \cdot W_{\varphi}(j\lambda_0) \quad (40)$$

Фильтры W_{ϑ} , W_{ψ} , W_{φ} синтезируются с учетом уровней шумов в измерительном тракте, квантования входной информации и допустимого запаздывания в трактах управления.

Командные сигналы на номерные сопловые блоки вычисляются следующим образом

$$\begin{aligned}
\delta_{14} = \delta_{13} &= \bar{\delta}_R + \delta_{\vartheta} + \delta_{\phi} \\
\delta_{12} = \delta_{11} &= \bar{\delta}_R + \delta_{\vartheta} - \delta_{\phi} \\
\delta_{22} = \delta_{21} &= \bar{\delta}_R - \delta_{\psi} + \delta_{\phi} \\
\delta_{23} = \delta_{24} &= \bar{\delta}_R - \delta_{\psi} - \delta_{\phi} \\
\delta_{33} = \delta_{34} &= \bar{\delta}_R - \delta_{\vartheta} + \delta_{\phi} \\
\delta_{31} = \delta_{32} &= \bar{\delta}_R - \delta_{\vartheta} - \delta_{\phi} \\
\delta_{42} = \delta_{41} &= \bar{\delta}_R + \delta_{\psi} + \delta_{\phi} \\
\delta_{44} = \delta_{43} &= \bar{\delta}_R + \delta_{\psi} - \delta_{\phi}
\end{aligned} \tag{41}$$

В случае, если $|\delta_{ij}| > \delta_{огр}$, то производится синхронное ”сжатие” сигналов угловой стабилизации

$$\begin{aligned}
\bar{\delta}_{ij} &= \bar{\delta}_R + (\pm\delta_{\vartheta} \pm \delta_{\varphi}) \cdot K_{comp} \\
\bar{\delta}_{ij} &= \bar{\delta}_R + (\pm\delta_{\psi} \pm \delta_{\varphi}) \cdot K_{comp}
\end{aligned} \tag{42}$$

Коэффициент ”сжатия” вычисляется следующим образом

$$K_{comp} = \begin{cases} 1, & \text{если } \max(\delta_{ij}) \leq D_{\Pi} \text{ и } \min(\delta_{ij}) \geq D_{\mathbf{M}} \\ \frac{D_{\Pi}}{\max(\delta_{ij})}, & \text{если } [\max(\delta_{ij}) \leq D_{\Pi} \text{ и } \min(\delta_{ij}) \geq D_{\mathbf{M}}] \end{cases} \tag{43}$$

10 Синтез алгоритмов стабилизации

10.1 Метод модального управления

Модальное управление - это методы формирования цепей обратных связей, придающих замкнутой системе заранее выбранное расположение корней характеристического уравнения.

Структура модального регулятора задается всегда одинаковой и представляет собой обратную связь по всем переменным состояния. Регулятор линейно преобразует поступившие сигналы, т.е. усиливает и суммирует эти сигналы x_i и выдает в качестве выходы их линейную комбинацию.

Это управление применяется, когда все составляющие вектора состояния объекта управления доступны непосредственному измерению (полная управляемость).

$$\dot{x} = Ax + Bu \quad (44)$$

Для получения желаемого быстродействия и характеристического уравнения системы введем линейную обратную связь по переменным состояния в соответствии с уравнением

$$u = v - Kx, \quad (45)$$

где v - новое обозначение вектора входных воздействий, u - вектор управляющих воздействий с выхода регулятора, K - матрица обратной связи.

K - является матрицей - строкой, элементы которой - есть коэффициенты обратных связей по всем составляющим вектора x .

$$K = [K_1 K_2 \dots K_n]$$

.

Структурная схема исходной системы с обратной связью по переменным состояния приведена на рис. (7)

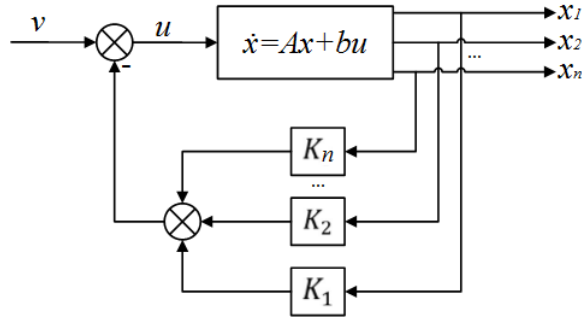


Рис. 7: Структурная схема системы с обратной связью по переменным состояния

Задача распределения корней характеристического уравнения замкнутой системы желаемым образом обеспечивает устойчивость заданной стабилизации и ее желаемую динамику быстрогодействия, перерегулирования и т.п.

Рассмотрим линейный объект $\dot{x} = Ax + Bu$ в форме пространства состояния.

В данном случае вектор x включает в себя параметры угла и угловой скорости соответственно по каналам управления, u - управление соответствующее приведенным моментам или силам действующее на летательный аппарат по рассматриваемому каналу.

Целью управления является

$$\|x(t) - x^*(t)\| \rightarrow 0, \quad t \rightarrow \infty,$$

где $x^*(t)$ - желаемая динамика замкнутой системы заданной эталонной модели.

Решение поставленной задачи методом модального управления находим в виде линейной обратной связи по составленному ОУ и заданному воздействию эталонной модели \dot{x}_* .

$$\dot{x}_* = A_* x_* + B_* u_*, \quad (46)$$

где

$$A_* = \begin{pmatrix} 0 & 1 \\ -\alpha_0^* & -\alpha_1^* \end{pmatrix} \quad B = \begin{pmatrix} 0 \\ \alpha_0^* \end{pmatrix}$$

Управляющее воздействие $u = Kx + K_r y_*$.

Компоненты вектора $K = \begin{pmatrix} K_1 \\ K_2 \end{pmatrix}$ и скалярного коэффициента усиления K_r определяется из соотношения:

$$BK = A_* - A \quad (47)$$

$$BK_r = B \quad (48)$$

Решение первого матричного уравнения существует если матрицы A и A^* согласованы по структуре, а ОУ является управляемым, то есть выполняется условие:

$$\text{rang}(B|AB|\dots|A^n B) = n \quad (49)$$

В рассматриваемом случае это условие соответствует выполнению следующих условий:

$$\det(B|AB) \neq 0 \quad (50)$$

Уравнения для определения K_r должно быть разрешенным относительно K_r при структурной согласованности B и B^* .

10.2 Постановка задачи слежения по ЦМ

Для того чтобы использовать вышеприведенный метод мы преобразуем исходную модель в форме пространства состояния с учетом того, что u_ξ , u_η , u_ζ - моменты по каналам.

$$\|x_i(t) - x_i^{\text{эм}}(t)\| \leq \bar{\Delta}_{xi}, \forall t \geq T_*, \quad (51)$$

где $x_i^{\text{эм}}$ - траектория эталонной модели заданной в виде:

$$\dot{x}_i^{\text{эм}} = A_{*i}x_i^{\text{эм}} + B_{*i}r_{*i}, \quad (52)$$

где $x_1 = (\xi\dot{\xi})^T$, $x_2 = (\eta\dot{\eta})^T$, $x_3 = (\zeta\dot{\zeta})^T$, $i = \overline{1,3}$, i - номер канала, u_i - управление.

$$u_1 = u_\xi, \quad u_2 = u_\eta, \quad u_3 = u_\zeta$$

$$r_{*1} = \xi_*(t), \quad r_{*2} = \eta_*(t), \quad r_{*3} = \zeta_*(t).$$

Приведенная выше формализация позволяет уйти от исходной системы к системе, описанной выше и использовать метод модального управления.

Желаемое качество отработки задается спектром матрицы эталонной модели: $\lambda_{*j}(A_{*i}) : \text{Re}\lambda_{*j} < 0$ (устойчивость), с заданным расположением корней.

Задаем расположение корней так, чтобы ближайший к мнимой оси полюс задавал быстроедействие системы, если он вещественный, то будет апериодическое движение, если будет пара ближайших к мнимой комплексно-сопряженных корней, то колебательное движение.

Исходя из того, что у нас система 2-го порядка выбирались 2 полюса так чтобы из требований времени второй полюс был отнесен дальше от мнимой оси, чтобы его динамика не сильно сказывалась

Точность определяем выбором параметром матрицы B_* так, чтобы статическая ошибка была равна 1.

11 Моделирование посадки ВА

В результате моделирования были получены графики требуемых траекторий, скоростей и ускорений (показано на рис. (8)), пространственная траектория полета ВА (показано на рис. (9)), требуемой траектории по высоте и требуемой траектории по высоте с учетом модального управления (показано на рис. (10)), отклонения от требуемой траектории с учетом управления (показано на рис. (11)).

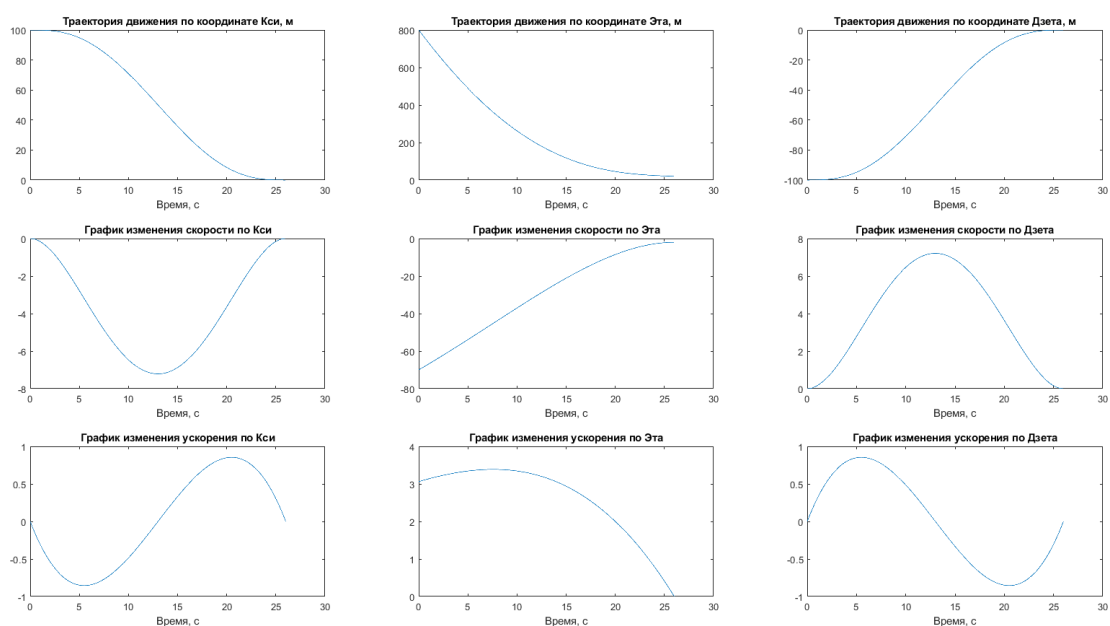


Рис. 8: Графики требуемых траекторий, скорости и ускорений

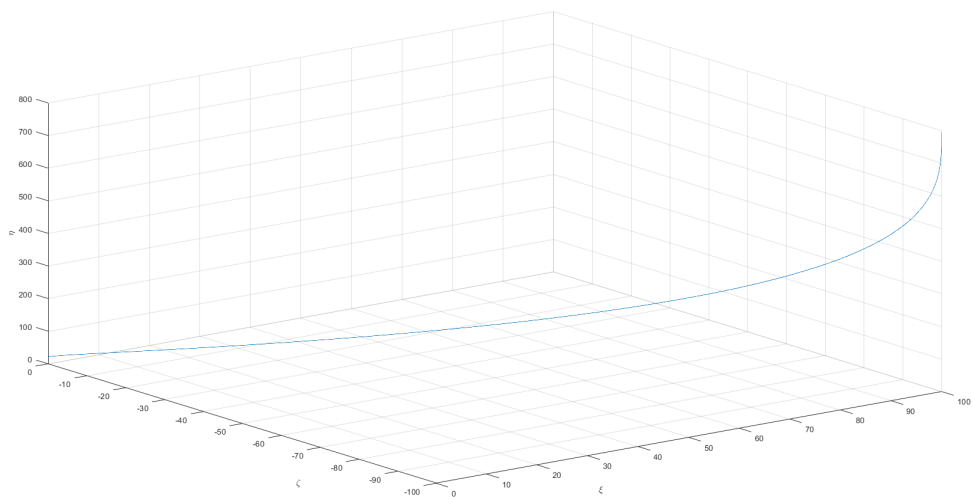


Рис. 9: Пространственная траектория полета ВА

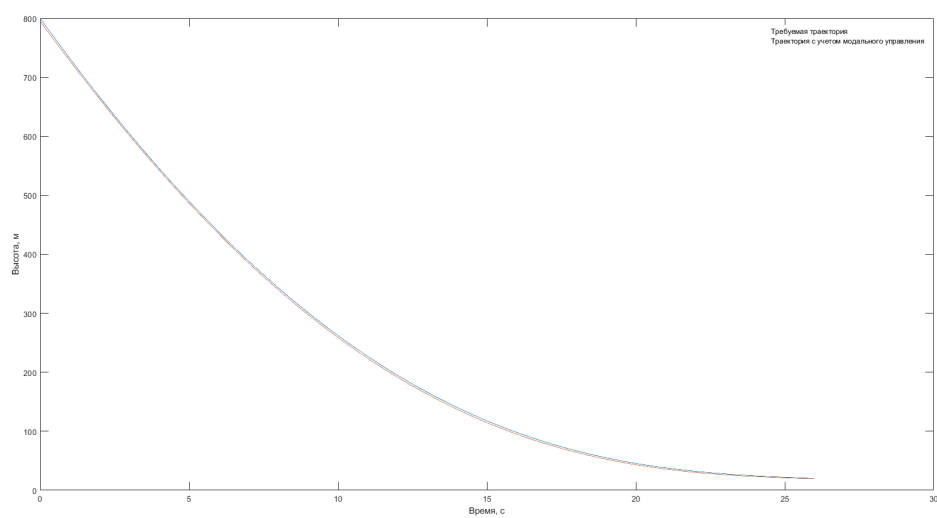


Рис. 10: Изменение высоты ВА с учетом модального управления и требуемое изменение высоты ВА

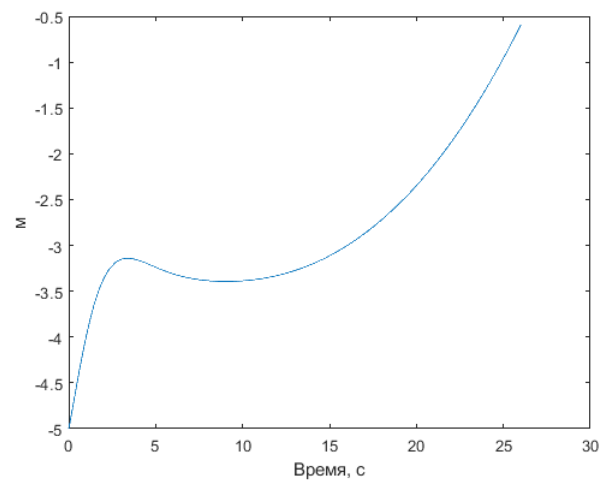


Рис. 11: Отклонение от требуемой траектории с учетом управления

ЗАКЛЮЧЕНИЕ

В данной работе была получена математическая модель движения центра масс возвращаемого аппарата (ВА), решена задача наведения, определена желаемая траектория, решена задача стабилизации по полученным желаемым траекториям, разработан алгоритм программного обеспечения для проведенного математического моделирования подтверждающий работоспособность предлагаемых схем проведения углового движения с желаемой динамикой.

Разработана программа для моделирования движения ВА на языке программирования MatLab.

Полученные результаты удовлетворяют заданным погрешностям.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

- [1] Дмитриевский А.А. «Внешняя баллистика.» - М.: Машиностроение, 1979.-479с.
- [2] Кузовков Н.Т. «Модальное управление и наблюдающие устройства.»- М.: «Машиностроение », 1976.- 184с.
- [3] Техническое задание на разработку СУ ПТДУ.
- [4] Крутько П.Д. «Обратные задачи динамики управляемых систем. Нелинейные модели».- М.: Наука, 1988.-328 с

ПРИЛОЖЕНИЕ

УТВЕРЖДЁН
А. В.000000 - 00 00 00 0

Программная масштабируемая модель имитации работы одноранговой сети

Спецификация

А. В.000000 - 00 00 00 – 0

Листов 3

Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

2020

Литера

[illegible]

Лист регистрации изменений

[illegible]

УТВЕРЖДЁН

А.В.00000 - 00 00 00 0 – 01 34 01

Синтез алгоритмов наведения и стабилизации центра масс возвращаемого
аппарата

Описание программы

А.В.00000 - 00 00 00 0 – 01 34 01

Листов 10

Инд. № подл	Подп. и дата
Взам. инв. №	Инд. № дубл.
Подп. и дата	Подп. и дата

2020

Литера

АННОТАЦИЯ

Данный документ содержит сведения о назначении программа синтеза алгоритмов наведения и стабилизации центра масс возвращаемого аппарата. Документ предоставляет данные о использованных технических и программных средствах.

СОДЕРЖАНИЕ

Оглавление

1. Общие сведения.....	4
2. Функциональное назначение	5
3. Описание логической структуры	6
4. Используемые технические средства.....	7
5. Вызов и загрузка	8
6. Входные данные	9
7. Выходные данные	10

1. ОБЩИЕ СВЕДЕНИЯ

Программа предназначена для моделирования пространственного движения летательного аппарата.

Для работы программы обязательно наличие установленной на компьютере среды разработки MatLab.

Программа разработана на языке MatLab.

2. ФУНКЦИОНАЛЬНОЕ НАЗНАЧЕНИЕ

1.1.Основные задачи программы

Программа предназначена для моделирования пространственного движения возвращаемого аппарата

1.2.Основные функции программы

Программа должна обеспечивать решение следующих задач:

- Формирование требуемых траекторий, скоростей, ускорений;
- Стабилизацию возвращаемого аппарата относительно требуемых траекторий;

3. ОПИСАНИЕ ЛОГИЧЕСКОЙ СТРУКТУРЫ

Функция `getCoefficient` рассчитывает коэффициенты полиномов для формирования требуемых траекторий, скоростей, ускорений.

Функция `getPathSpeedBoost` формирует векторы требуемых траекторий, скоростей, ускорений

Далее будет приведена схема алгоритма данной программы, показанная на рис.1.



Рис.1 Схема алгоритма программы

4. Используемые технические средства

а. Состав аппаратных средств

Программа рассчитана на работу с ЭВМ, имеющей:

- манипулятор «мышь»;
- клавиатуру;
- не менее 256 Мбайт ОЗУ;
- центральный процессор архитектуры x86-64: Intel Core 2 Duo CPU E6550 или мощнее;
- цветной монитор с разрешением не менее 1280x1024.

б. Состав программных средств

Программа рассчитана на работу под ОС Windows 10.

5. Вызов и загрузка

Для вызова программы необходимо открыть все файлы в среде разработки MatLab, а затем запустить файл «SpaceCraft.m».

6. Входные данные

В качестве входных данных подаются начальные координаты возвращаемого аппарата, координаты точки зависания возвращаемого аппарата, время полета, ускорение свободного падения.

7. Выходные данные

Выходные данные представляются в виде графиков, выводимых на экран.

УТВЕРЖДЁН

А.В.00000 - 00 00 00 0 – 0 32 01

Синтез алгоритмов наведения и стабилизации центра масс возвращаемого аппарата

Текст программы

А.В.00000 - 00 00 00 0 – 0 32 01

Листов 14

Инв. Nподл.	Подп. и дата	Взаим инв. N	ИнвNдубл	Подп. и дата

2020

Литера

СОДЕРЖАНИЕ

1. Общие сведения.....	3
1.1. Среда разработки.....	3
1.2. Среда выполнения.....	3
2. Текст программы.....	4
2.1. Основные файлы программы.....	4
2.1.1. Файл SpaceCraft.m	4
2.1.2. Файл getCoefficient.m	13
2.1.3. Файл getPathSpeedBoost.m.....	13

1. ОБЩИЕ СВЕДЕНИЯ

1.1. Среда разработки

Программа разработана в среде MatLab R2019b.

1.2. Среда выполнения

Программа предназначена для запуска в ОС Windows 10

2. ТЕКСТ ПРОГРАММЫ

2.1. Основные файлы программы

2.1.1. Файл SpaceCraft.m

```

clc;
clear variables;
close all;

%% Начальные условия
T = 26;
g = -9.8;
dt = 0.01;
t = 0:dt:T;
N = numel(t);
C = [0 1];
D = 0;
B = [0; 1];
l1=-8;
l2=-15;

A=[
    0 1;
    1 1
];

% sys = ss(A, B, C, D)
% sys2 = mksys(A, B, C, D);

% Массивы СВСК
FX1 = zeros(1,N); % суммарная сила, действующая относительно X1 [Н]
FY1 = zeros(1,N); % суммарная сила, действующая относительно Y1 [Н]
FZ1 = zeros(1,N); % суммарная сила, действующая относительно Z1 [Н]
FuprY1 = zeros(1,N); % управляющая сила относительно Y1 [Н]
FuprZ1 = zeros(1,N); % управляющая сила относительно Z1 [Н]
Xc1 = zeros(1,N); % аэродинам. сила относительно X1 [Н]
Yc1 = zeros(1,N); % аэродинам. сила относительно Y1 [Н]
Zc1 = zeros(1,N); % аэродинам. сила относительно Z1 [Н]
Teta = zeros(1,N); % угол [рад]
Fi = zeros(1,N);
Psi = zeros(1,N);
OmegaX1 = zeros(1,N); % угловая скорость относительно X1 [рад/с]
OmegaY1 = zeros(1,N); % угловая скорость относительно Y1 [рад/с]
OmegaZ1 = zeros(1,N); % угловая скорость относительно Z1 [рад/с]
MX1 = zeros(1,N); % суммарный момент относительно X1 [Н*м]
MY1 = zeros(1,N); % суммарный момент относительно Y1 [Н*м]
MZ1 = zeros(1,N); % суммарный момент относительно Z1 [Н*м]
MuprX1 = zeros(1,N); % управляющий момент относительно X1 [Н*м]
MuprY1 = zeros(1,N); % управляющий момент относительно Y1 [Н*м]
MuprZ1 = zeros(1,N); % управляющий момент относительно Z1 [Н*м]
MaY1 = zeros(1,N); % аэродинам. момент относительно Y1 [Н*м]
MaZ1 = zeros(1,N); % аэродинам. момент относительно Z1 [Н*м]
McmY1 = zeros(1,N); % момент смещения ЦМ относительно Y1 [Н*м]
McmZ1 = zeros(1,N); % момент смещения ЦМ относительно Z1 [Н*м]

```

```

Rsum = zeros(1,N); % суммарная тяга ПТДУ [Н]
R_TP = zeros(1,N); % требуемая тяга [Н]
dTeta = zeros(1,N);
dPsi = zeros(1,N);
dFi = zeros(1,N);
PsiTP = zeros(1,N);
TetaTP = zeros(1,N);
X1TP = zeros(1,N);
Y1TP = zeros(1,N);
Z1TP = zeros(1,N);
VX1TP = zeros(1,N);
VY1TP = zeros(1,N);
VZ1TP = zeros(1,N);
WX1TP = zeros(1,N);
WY1TP = zeros(1,N);
WZ1TP = zeros(1,N);
KsiTP = zeros(1,N);
VksiTP = zeros(1,N);
WksiTP = zeros(1,N);
EtaTP = zeros(1,N);
VetaTP = zeros(1,N);
WetaTP = zeros(1,N);
DzetaTP = zeros(1,N);
VdzetaTP = zeros(1,N);
WdzetaTP = zeros(1,N);
%-----НУ и коэффициенты-----
ro = 1.25; % плотность атмосферы [кг*сек^2/м^4]
m0 = 7770; % начальная масса аппарата [кг]
Sm = 15.2; % площадь мидела аппарата [м^2]
Yt = 0.2; % смещение ЦМ относительно оси X1 [м]
Zt = 0; % смещение ЦМ относительно оси Z1 [м]
L = 3.9; % длина аппарата [м]
Ixx = 19000; % момент инерции вокруг оси X [кг*м^2]
Iyy = 17000; % момент инерции вокруг оси Y [кг*м^2]
Izz = 17000; % момент инерции вокруг оси Z [кг*м^2]
g = 9.81; % ускорение свободного падения [м/с^2]
R_TP_min = 6500*9.81; % минимальная требуемая тяга [Н]
R_TP_max = 24000*9.81; % максимальная требуемая тяга [Н]
% параметры сопел
alphaC = deg2rad(5.73); % угол наклона сопел к продольной оси X [рад]
lc = 0.4; % плечо сопел [м]
Loy = 2;
Lx = Loy*cot(alphaC);
Xt = 1.35;
lx = Lx - Xt;
Lc = lx*sin(alphaC);
deltasummax = deg2rad(480);
Kteta = 0.16*57.3;
Kpsi = 0.16*57.3;
Kfi = 5*57.3;
Kw = 0;
Kwt = 0;
tauP = 0.4;
tauB = 0.4;
% углы
Fi(1,1) = 0; % [рад]
Psi(1,1) = 0; % [рад]
Teta(1,1) = -pi/2; % [рад]
Fik = 0; % [рад]
Psik = 0; % [рад]

```

```

Tetak = -pi/2; % [рад]
PsiTP(1,1) = 0;
TetaTP(1,1) = Tetak;
% Расчет НУ в СВСК
a11 = cos(Teta(1,1))*cos(Psi(1,1));
a12 = cos(Teta(1,1))*sin(Psi(1,1))*sin(Fi(1,1)) - sin(Teta(1,1))*cos(Fi(1,1));
a13 = cos(Teta(1,1))*sin(Psi(1,1))*cos(Fi(1,1)) + sin(Teta(1,1))*sin(Fi(1,1));
a21 = sin(Teta(1,1))*cos(Psi(1,1));
a22 = sin(Teta(1,1))*sin(Psi(1,1))*sin(Fi(1,1)) + cos(Teta(1,1))*cos(Fi(1,1));
a23 = sin(Teta(1,1))*sin(Psi(1,1))*cos(Fi(1,1)) - cos(Teta(1,1))*sin(Fi(1,1));
a31 = -sin(Psi(1,1));
a32 = cos(Psi(1,1))*sin(Fi(1,1));
a33 = cos(Psi(1,1))*cos(Fi(1,1));

%% Задание полюсов
% p = [-8; -15];
% pe = 2*[-8; -7];
% K = place(A, B, p);
% L = acker(A, B, pe)';

%% Формирование наблюдателя с выбранными полюсами
% newsys = append(sys, K)
% rsys = reg(sys, K, L);
% sys1 = parallel(sys, rsys, [1], [1], [2]);
% sys2 = connect(sys1, );

%% Проверка управляемости и наблюдаемости
% Co = ctrb(A, B);
% Unco = length(A) - rank(Co);
% Do = obsv(sys);
% Undo = length(A) - rank(Do);

%% Расчет траектории по эта, скорости и ускорения с учётом алгоритмов
описывающих функций

% Начальные условия по координате Кси
xiPathInit = 100;
xiSpeedInit = 0;
xiBoostInit = 0;

% Конечные условия по координате Кси
xiPathFin = 0;
xiSpeedFin = 0;
xiBoostFin = 0;

% Получение коэффициентов по координате Кси
[coefXi0, coefXi1, coefXi2, coefXi3, coefXi4, coefXi5] =
getCoefficient(xiPathInit, xiSpeedInit, xiBoostInit, xiPathFin, xiSpeedFin,
xiBoostFin, T);

% pathXi - траектория пути по Кси
% speedXi - скорость по Кси
% boostXi - ускорения по Кси
[pathXi, speedXi, boostXi] = getPathSpeedBoost(coefXi0, coefXi1, coefXi2,
coefXi3, coefXi4, coefXi5, t, N);

% Начальные условия по координате Эта
etaPathInit = 800;

```

A. B.00000 – 00 00 00 0-01 32 01

```

etaSpeedInit = -70;
etaBoostInit = etaSpeedInit ^ 2 / (2 * etaPathInit);

% Конечные условия по координате Эта
etaPathFin = 20;
etaSpeedFin = -2;
etaBoostFin = 0;

% Получение коэффициентов по координате Эта
[coefEta0, coefEta1, coefEta2, coefEta3, coefEta4, coefEta5] =
getCoefficient(etaPathInit, etaSpeedInit, etaBoostInit, etaPathFin, etaSpeedFin,
etaBoostFin, T);

% pathEta - траектория пути по Эта
% speedEta - скорость по Эта
% boostEta - ускорения по эта
[pathEta, speedEta, boostEta] = getPathSpeedBoost(coefEta0, coefEta1, coefEta2,
coefEta3, coefEta4, coefEta5, t, N);

% Начальные условия по координат Дзета
zetaPathInit = -100;
zetaSpeedInit = 0;
zetaBoostInit = 0;

% Конечные условия по координате Дзета
zetaPathFin = 0;
zetaSpeedFin = 0;
zetaBoostFin = 0;

% Получение коэффициентов по координате Дзета
[coefZeta0, coefZeta1, coefZeta2, coefZeta3, coefZeta4, coefZeta5] =
getCoefficient(zetaPathInit, zetaSpeedInit, zetaBoostInit, zetaPathFin,
zetaSpeedFin, zetaBoostFin, T);

% pathZeta - траектория пути по Кси
% speedZeta - скорость по Кси
% boostZeta - ускорения по Кси
[pathZeta, speedZeta, boostZeta] = getPathSpeedBoost(coefZeta0, coefZeta1,
coefZeta2, coefZeta3, coefZeta4, coefZeta5, t, N);

%% Моделирование системы по желаемой траектории
% f0 = [y01 y0]
% [f, t, h] = lsim(sys, y, t, f0);
% [freg, t, hreg] = lsim(rsys, y, t, f0);

for i=1:N
    X1TP(1,i) = a11*pathXi(1,i) + a21*pathEta(1,i) + a31*pathZeta(1,i);
    VX1TP(1,i) = a11*speedXi(1,i) + a21*speedEta(1,i) + a31*speedZeta(1,i);
    WX1TP(1,i) = a11*boostXi(1,i) + a21*boostEta(1,i) + a31*boostZeta(1,i);

    Y1TP(1,i) = a12*pathXi(1,i) + a22*pathEta(1,i) + a32*pathZeta(1,i);
    VY1TP(1,i) = a12*speedXi(1,i) + a22*speedEta(1,i) + a32*speedZeta(1,i);
    WY1TP(1,i) = a12*boostXi(1,i) + a22*boostEta(1,i) + a32*boostZeta(1,i);

    Z1TP(1,i) = a13*pathXi(1,i) + a23*pathEta(1,i) + a33*pathZeta(1,i);

```



```

VZ1TP(1,i) = a13*speedXi(1,i) + a23*speedEta(1,i) + a33*speedZeta(1,i);
WZ1TP(1,i) = a13*boostXi(1,i) + a23*boostEta(1,i) + a33*boostZeta(1,i);
end

for i=1:N-1
    % Пересчет матрицы направляющих косинусов
    a11 = cos(Teta(1,i))*cos(Psi(1,i));
    a12 = cos(Teta(1,i))*sin(Psi(1,i))*sin(Fi(1,i)) -
sin(Teta(1,i))*cos(Fi(1,i));
    a13 = cos(Teta(1,i))*sin(Psi(1,i))*cos(Fi(1,i)) +
sin(Teta(1,i))*sin(Fi(1,i));
    a21 = sin(Teta(1,i))*cos(Psi(1,i));
    a22 = sin(Teta(1,i))*sin(Psi(1,i))*sin(Fi(1,i)) +
cos(Teta(1,i))*cos(Fi(1,i));
    a23 = sin(Teta(1,i))*sin(Psi(1,i))*cos(Fi(1,i)) -
cos(Teta(1,i))*sin(Fi(1,i));
    a31 = -sin(Psi(1,i));
    a32 = cos(Psi(1,i))*sin(Fi(1,i));
    a33 = cos(Psi(1,i))*cos(Fi(1,i));

    % Требуемая ориентация вектора тяги eR
    Wksi_TP = boostXi(1,i);
    Weta_TP = boostEta(1,i) + g;
    Wdzeta_TP = boostZeta(1,i);
    W_TP = (Wksi_TP^2 + Weta_TP^2 + Wdzeta_TP^2)^0.5;

    eRksi = - Wksi_TP / W_TP;
    eReta = - Weta_TP / W_TP;
    eRdzeta = - Wdzeta_TP / W_TP;

    % eR - требуемое направление оси X1 (против направления тяги)
    PsiTP(1,i) = - asin(eRdzeta);

    if eRksi >= 0 && eReta < 0
        TetaTP(1,i) = - acos(eRksi);
    end
    if eRksi < 0 && eReta < 0
        TetaTP(1,i) = - acos(eRksi);
    end

    if TetaTP(1,i) < -deg2rad(110)
        TetaTP(1,i) = -deg2rad(110);
    end
    if TetaTP(1,i) > -deg2rad(70)
        TetaTP(1,i) = -deg2rad(70);
    end
    if PsiTP(1,i) < -deg2rad(20)
        PsiTP(1,i) = -deg2rad(20);
    end
    if PsiTP(1,i) > deg2rad(20)
        PsiTP(1,i) = deg2rad(20);
    end
    dPsi(1,i) = Psi(1,i) - PsiTP(1,i);
    dTeta(1,i) = Teta(1,i) - TetaTP(1,i);
    dFi(1,i) = Fi(1,i);
    deltaP = Kpsi*dPsi(1,i) + Kpsi*tauP*OmegaY1(1,i);
    deltaT = Kteta*dTeta(1,i) + Kteta*tauP*OmegaZ1(1,i);
    deltaB = Kfi*dFi(1,i) + Kfi*tauB*OmegaX1(1,i);

```

```

% требуемая тяга
R_TP_X1 = m0 * W_TP;
R_TP(1,i) = R_TP_X1 / 0.7;
if R_TP(1,i) > R_TP_max
    R_TP(1,i) = R_TP_max;
end
if R_TP(1,i) < R_TP_min
    R_TP(1,i) = R_TP_min;
end

deltaR = 0;
if (R_TP(1,i) > 63765 && R_TP(1,i) < 147150)
    deltaR = deg2rad(36);
end
if (R_TP(1,i) > 147150 && R_TP(1,i) < 235440)
    deltaR = deg2rad(30);
end
if R_TP(1,i) == 235440
    deltaR = deg2rad(24);
end

delta1 = deltaR + deltaT + deltaB;
delta2 = deltaR + deltaT - deltaB;
delta3 = deltaR - deltaP + deltaB;
delta4 = deltaR - deltaP - deltaB;
delta5 = deltaR - deltaT + deltaB;
delta6 = deltaR - deltaT - deltaB;
delta7 = deltaR + deltaP + deltaB;
delta8 = deltaR + deltaP - deltaB;
R1 = R_TP(1,i)*delta1/deltasummax;
R2 = R_TP(1,i)*delta2/deltasummax;
R3 = R_TP(1,i)*delta3/deltasummax;
R4 = R_TP(1,i)*delta4/deltasummax;
R5 = R_TP(1,i)*delta5/deltasummax;
R6 = R_TP(1,i)*delta6/deltasummax;
R7 = R_TP(1,i)*delta7/deltasummax;
R8 = R_TP(1,i)*delta8/deltasummax;
if ((abs(VZ1TP(1,i))<=0.1) && (VY1TP(1,i)<=0))
    Gamma = 0;
end
if ((abs(VZ1TP(1,i))<=0.1) && (VY1TP(1,i)>0))
    Gamma = deg2rad(180);
end
if VZ1TP(1,i)>0.1
    Gamma = atan(VY1TP(1,i)/VZ1TP(1,i)) + deg2rad(90);
end
if VZ1TP(1,i)<-0.1
    Gamma = atan(VY1TP(1,i)/VZ1TP(1,i)) + deg2rad(270);
end

if i>2
    al = atan(-VY1TP(1,i)/VX1TP(1,i)); % угол атаки
    bet = atan(VZ1TP(1,i)/VX1TP(1,i)); % угол скольжения
else
    al = 0;
    bet = 0;
end

```

A. B.00000 – 00 00 00 0-01 32 01

```
V = (VX1TP(1,i)^2 + VY1TP(1,i)^2 + VZ1TP(1,i)^2)^0.5;
q = ro*V^2/2;
```

```
if al >= 0 && al < 10
    Cx = 0.812;
    Cy = 0.024;
    mz = 0.03;
end
if al >= 10 && al < 20
    Cx = 0.839;
    Cy = 0.026;
    mz = 0.0;
end
if al >= 20 && al < 30
    Cx = 0.776;
    Cy = -0.024;
    mz = -0.029;
end
if al >= 30 && al < 40
    Cx = 0.69;
    Cy = 0.016;
    mz = -0.05;
end
if al >= 40 && al < 50
    Cx = 0.571;
    Cy = 0.09;
    mz = -0.06;
end
if al >= 50 && al < 60
    Cx = 0.48;
    Cy = 0.25;
    mz = -0.04;
end
if al >= 60 && al < 70
    Cx = 0.38;
    Cy = 0.38;
    mz = -0.02;
end
if al >= 70 && al < 80
    Cx = 0.28;
    Cy = 0.37;
    mz = -0.013;
end
if al >= 80 && al < 90
    Cx = 0.08;
    Cy = 0.34;
    mz = -0.009;
end
if al >= 90
    mz = 0;
end
```

```
MaY1(1,i) = mz*q*Sm*L*sign(bet);
MaZ1(1,i) = mz*q*Sm*L*sign(al);
McmY1(1,i) = R_TP(1,i)*Zt;
McmZ1(1,i) = -R_TP(1,i)*Yt;
MuprX1(1,i) = (R2 + R4 + R6 + R8 - R1 - R3 - R5 - R7)*sin(alphaC)*lc;
MuprY1(1,i) = (R3 + R4 - R7 - R8)*Lc;
MuprZ1(1,i) = (R5 + R6 - R1 - R2)*Lc;
```

A. B.00000 – 00 00 00 0-01 32 01

```

FuprY1(1, i) = -1 * (R5 + R6 - R1 - R2)*sin(alphaC);
FuprZ1(1, i) = (R3 + R4 - R7 - R8)*sin(alphaC);
MX1(1,i) = MuprX1(1,i);
MY1(1,i) = MuprY1(1,i) + 0*MaY1(1,i) + 0*McmY1(1,i);
MZ1(1,i) = MuprZ1(1,i) + 0*MaZ1(1,i) + 0*McmZ1(1,i);
Teta(1,i+1) = Teta(1,i) +
dt*(1/cos(Psi(1,i))*(OmegaY1(1,i)*sin(Fi(1,i))+OmegaZ1(1,i)*cos(Fi(1,i))));
Psi(1,i+1) = Psi(1,i) + dt*(OmegaY1(1,i)*cos(Fi(1,i))-
OmegaZ1(1,i)*sin(Fi(1,i)));
Fi(1,i+1) = Fi(1,i) +
dt*(OmegaX1(1,i)+tan(Psi(1,i))*(OmegaY1(1,i)*sin(Fi(1,i))+OmegaZ1(1,i)*cos(Fi(1,
i))));
OmegaX1(1,i+1) = OmegaX1(1,i) + dt*MX1(1,i)/Ixx;
OmegaY1(1,i+1) = OmegaY1(1,i) + dt*MY1(1,i)/Iyy;
OmegaZ1(1,i+1) = OmegaZ1(1,i) + dt*MZ1(1,i)/Izz;
end

[coefEta0, coefEta1, coefEta2, coefEta3, coefEta4, coefEta5] =
getCoefficient(etaPathInit-5, etaSpeedInit+1, etaBoostInit, etaPathFin,
etaSpeedFin, etaBoostFin, T);

% pathEta - траектория пути по Эта
% speedEta - скорость по Эта
% boostEta - ускорения по эта
[pathEtaBias, speedEtaBias, boostEtaBias] = getPathSpeedBoost(coefEta0,
coefEta1, coefEta2, coefEta3, coefEta4, coefEta5, t, N);
p = [-0.01; -0.5];
pe = 2*[-8; -7];
K = place(A, B, p);
u = zeros(1, N);
Kr = p(1, 1) * p(2, 1) *0.001;
% K(1, 1) = 0.5;
% K(1, 2) = 2.5;
for i = 1:N-1
    u(1, i) = - K(1, 1) * (pathEtaBias(1, i) - pathEta(1, i)) - K(1, 2) *
(speedEtaBias(1, i) - speedEta(1, i)) +Kr*pathEta(1, i);
    speedEtaBias(1, i+1) = speedEtaBias(1, i) + dt*u(1, i);
    pathEtaBias(1, i+1) = pathEtaBias(1, i) + dt*speedEtaBias(1, i);
end

% Графики траекторий
% Траектория по Кси
figure('Name', 'Траектория движения по координате Кси, м');
subplot(3, 3, 1);
plot(t, pathXi);
title('Траектория движения по координате Кси, м');
xlabel('Время, с');

% Траектория по Эта
subplot(3, 3, 2);
plot(t, pathEta);
title('Траектория движения по координате Эта, м')
xlabel('Время, с');

% Траектория по Дзета
subplot(3, 3, 3);
plot(t, pathZeta);
title('Траектория движения по координате Дзета, м')

```

```

xlabel('Время, с');

%   Трехмерный график
% subplot(3, 3, 4);
% plot3(pathZeta, pathXi, pathEta);

%%   Графики изменения скорости
%   Скорость по Кси
subplot(3, 3, 4);
plot(t, speedXi);
title('График изменения скорости по Кси');
xlabel('Время, с');

%   Скорость по Эта
subplot(3, 3, 5);
plot(t, speedEta);
title('График изменения скорости по Эта');
xlabel('Время, с');

%   Скорость по Дзета
subplot(3, 3, 6);
plot(t, speedZeta);
title('График изменения скорости по Дзета');
xlabel('Время, с');

%%   Графики изменения ускорения
%   Ускорение по Кси
subplot(3, 3, 7);
plot(t, boostXi);
title('График изменения ускорения по Кси');
xlabel('Время, с');

%   Ускорение по Эта
subplot(3, 3, 8);
plot(t, boostEta);
title('График изменения ускорения по Эта');
xlabel('Время, с');

%   Ускорение по Дзета
subplot(3, 3, 9);
plot(t, boostZeta);
title('График изменения ускорения по Дзета');
xlabel('Время, с');

%%
% Тректория с учетом модального управления
figure('Name', 'Тректория с учетом модального управления')
plot(t, pathEta);
ylabel('Высота, м')
xlabel('Время, с');
hold on;
plot(t, pathEtaBias)
legend('Требуемая траектория', 'Траектория с учетом модального управления')
ylabel('Высота, м')
xlabel('Время, с');

% Графики управлений
figure('Name', 'управлений')
plot(t, u);

```

```
% График ошибки
figure('Name', 'Отклонение от требуемой траектории')
plot(t, (pathEtaBias - pathEta))
ylabel('м')
xlabel('Время, с');
```

```
% Пространственная траектория
figure('Name', 'Пространственная траектория')
plot3(pathXi, pathZeta, pathEtaBias);
grid on;
hold all;
xlabel('\xi');
ylabel('\zeta');
zlabel('\eta');
```

2.1.2. Файл getCoefficient.m

```
function [coef0, coef1, coef2, coef3, coef4, coef5] = getCoefficient(pathInit,
speedInit, boostInit, pathFin, speedFin, boostFin, T)
%UNTITLED2 Summary of this function goes here
% Detailed explanation goes here
coef0 = pathInit;
coef1 = speedInit;
coef2 = boostInit/2;
coef3 = boostFin / (2 * T) - (6 * speedInit) / T ^ 2 - (3 * boostInit) / (2*T)
- (4*speedFin)/T^2 - (10*pathInit)/T^3 + (10*pathFin)/T^3;
coef4 = (15 * pathInit)/T^4 + (8 * speedInit) / T ^ 3 + (3 *
boostInit)/(2*T^2) + (7*speedFin)/T^3 - boostFin/T^2 - (15*pathFin)/T ^ 4;
coef5 = boostFin / (2 * T ^ 3) - (3 * speedInit) / T ^ 4 - boostInit / (2 *
T ^ 3) - (3 * speedFin) / T ^ 4 - (6 * pathInit) / T ^ 5 + (6 * pathFin) / T ^
5;
end
```

2.1.3. Файл getPathSpeedBoost.m

```
function [path, speed, boost] = getPathSpeedBoost(a0, a1, a2, a3, a4, a5, t, N)
%Функция getPathSpeedBoost принимает в качестве аргументов коэффициенты
%a0-a5, t - вектор времени, N - количество точек
path = zeros(1, N);
speed = zeros(1, N);
boost = zeros(1, N);
for i = 1:N
    path(1, i) = a0 + a1*t(1, i) + a2*t(1, i)^2 + a3*t(1, i)^3 + a4*t(1,
i)^4 + a5*t(1, i)^5;
    speed(1, i) = a1 + 2*a2*t(1, i) + 3*a3*t(1, i)^2 + 4*a4*t(1, i)^3 +
5*a5*t(1, i)^4;
    boost(1, i) = 2*a2 + 6*a3*t(1, i) + 12*a4*t(1, i)^2 + 20*a5*t(1, i)^3;
end
end
```

[illegible]