

МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение
высшего образования

**«САРАТОВСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ИМЕНИ Н.Г.ЧЕРНЫШЕВСКОГО»**

Кафедра Математического и компьютерного моделирования

**Оптимальное управление разворотом осесимметричного
космического аппарата**

АВТОРЕФЕРАТ БАКАЛАВРСКОЙ РАБОТЫ

студента 4 курса 411 группы

направление 01.03.02 — Прикладная математика и информатика

механико-математического факультета

Лисюткина Михаила Юрьевича

Научный руководитель

доцент, к. т. н., доцент

И. А. Панкратов

Зав. кафедрой

зав. каф., д.ф.-м.н., доцент

Ю. А. Блинков

Саратов 2023

Введение. В данной работе рассмотрена задача оптимального в смысле комбинированного функционала качества разворота осесимметричного космического аппарата (КА).

Актуальность.

Исследованию задачи оптимального управления переориентацией твердого тела в различных постановках посвящено множество публикаций. В частности, в работе К. Б. Алексеева рассматриваются вопросы оптимального разворота КА по быстродействию и минимуму энергетических затрат. Аналитическое решение было получено принципом максимума Л.С. Понтрягина для случая, когда область допустимых значений управляющего момента ограничена сферой, а сам КА разворачивается вокруг вектора конечного поворота. В работах Н. Е. Зубова оптимальные управления находились методом совмещенного синтеза на основе алгоритма с прогнозирующей моделью, причем минимизировался функционал обобщенной работы.

Целью данной работы является реализация исследований, проведенных в статье Я. Г. Сапункова и А. В. Молоденкова с использованием символьной алгебры. Под пространственной переориентацией будем понимать перевод связанных с корпусом КА осей $OXYZ$ из одного известного углового положения в другое известное (обычно заданное) угловое положение за конечное время $t_{\text{кон}}$.

Структура бакалаврской работы.

Работа состоит из четырех разделов и приложения.

— В первом разделе даны теоретические основы алгебры кватернионов: сложение, умножение, норма кватерниона и т.д.

— Во втором разделе описан общий вид постановки задач оптимального управления, а также их способ решения, впервые представленный Л. С. Понтрягиным.

— В третьем разделе дано описание задачи оптимального в смысле комбинированного функционала разворота осесимметричного космического аппарата.

— В четвертой главе рассмотрено решение данной задачи методом компьютерной алгебры на языке программирования *Python*. С использованием символьной алгебры и принципа максимума Понтрягина найдена аналитиче-

ская запись дифференциальных уравнений основной и сопряженной систем. Затем с использованием пакета *scipy* строится вектор невязок, находятся его нули и решается исходная краевая задача.

— В приложении представлен исходный код программы

Основное содержание работы.

Движение КА как твердого тела с одной осью симметрии вокруг центра масс описывается дифференциальными уравнениями:

$$\begin{aligned} 2\dot{\mathbf{L}} &= \mathbf{L} \circ \mathbf{w}, \\ I_1\dot{w}_1 &= M_1, \\ I_2\dot{w}_2 &= M_2 - (I_1 - I_2)w_1w_3, \\ I_2\dot{w}_3 &= M_3 + (I_1 - I_2)w_1w_2. \end{aligned} \tag{1}$$

$\mathbf{L}(t)$ — нормированный кватернион поворота КА:

$$\begin{aligned} \mathbf{L}(t) &= l_0(t) + l_1(t)\mathbf{i}_1 + l_2(t)\mathbf{i}_2 + l_3(t)\mathbf{i}_3 = [l_0(t), l_1(t), l_2(t), l_3(t)]^T, \\ \|\mathbf{L}\| &= l_0^2 + l_1^2 + l_2^2 + l_3^2 = 1, \end{aligned}$$

$\mathbf{w}(t)$ — вектор угловой скорости КА:

$$\mathbf{w}(t) = w_1(t)\mathbf{i}_1 + w_2(t)\mathbf{i}_2 + w_3(t)\mathbf{i}_3 = [w_1(t), w_2(t), w_3(t)]^T,$$

$\mathbf{M}(t)$ — вектор управления:

$$\mathbf{M}(t) = M_1(t)\mathbf{i}_1 + M_2(t)\mathbf{i}_2 + M_3(t)\mathbf{i}_3 = [M_1(t), M_2(t), M_3(t)]^T.$$

I_1, I_2 — главные центральные моменты инерции твердого тела,

$$I_1, I_2 = \text{const} > 0.$$

Заданы произвольные граничные условия по угловому положению

$$\mathbf{L}(0) = \mathbf{L}_0, \quad \mathbf{L}(T) = \mathbf{L}_T \tag{2}$$

и угловой скорости КА

$$\mathbf{w}(0) = \mathbf{w}_0, \quad \mathbf{w}(T) = \mathbf{w}_T. \quad (3)$$

Требуется определить оптимальное управление $\mathbf{M}^{\text{опт}}(t)$ системой (1), при условиях (2)-(3), доставляющее минимум функционалу

$$J = \int_0^T (1 + a\mathbf{M}^2) dt, \quad (4)$$

где $a > 0$. Функционал (4) представляет собой комбинацию двух критериев с размерным весовым множителем a : времени и энергии, затраченных на разворот КА.

В ходе преобразований исходная система (1) приняла следующий вид:

$$\begin{aligned} 2\dot{\Lambda} &= \Lambda \circ \boldsymbol{\omega}, \\ \dot{\boldsymbol{\omega}} &= \mathbf{u}, \\ \dot{\boldsymbol{\theta}} &= b_2\omega_1, \\ \boldsymbol{\theta}(0) &= 0, \end{aligned} \quad (5)$$

$$\begin{aligned} \boldsymbol{\omega}(0) &= \boldsymbol{\omega}_0 = b_1w_{0_1}\mathbf{i}_1 + w_{0_2}\mathbf{i}_2 + w_{0_3}\mathbf{i}_3, \quad \Lambda(0) = \Lambda_0 = \mathbf{L}_0, \\ \boldsymbol{\omega}(T) &= \boldsymbol{\omega}_T = \tilde{\mathbf{B}}(\boldsymbol{\theta}(T)) \circ (b_1w_{T_1}\mathbf{i}_1 + w_{T_2}\mathbf{i}_2 + w_{T_3}\mathbf{i}_3) \circ \mathbf{B}(\boldsymbol{\theta}(T)), \\ \Lambda(T) &= \Lambda_T = L_T \circ \mathbf{B}(\boldsymbol{\theta}(T)), \end{aligned}$$

$$J = \int_0^T (1 + I_2^2 \mathbf{u}^2) dt \rightarrow \min \quad (6)$$

где $w_{0_1}, w_{0_2}, w_{0_3}, w_{T_1}, w_{T_2}, w_{T_3}$ - компоненты вектора

$$\mathbf{w}(t) = [w_1(t), w_2(t), w_3(t)]^T,$$

T — неизвестное время окончания процесса,

$\Lambda(t)$ — безразмерный кватернион ориентации,

$\boldsymbol{\omega}(t)$ — безразмерный вектор угловой скорости,

$\boldsymbol{\theta}(t)$ — функция изменения угла поворота космического аппарата.

Данную задачу можно переформулировать так: в восьмимерном фазовом пространстве $\Lambda \times \boldsymbol{\omega} \times \boldsymbol{\theta}$ управляемую систему (5) необходимо оптимально в

смысле функционала (6) перевести из начального состояния на многообразии, которое определяется соотношениями

$$\begin{aligned} \text{vect} \left(\mathbf{\Lambda} \circ \tilde{\mathbf{B}}(\theta) \circ \tilde{\mathbf{L}}_T \right) &= 0, \\ \boldsymbol{\omega} - \tilde{\mathbf{B}}(\theta) \circ (b_1 w_{T_1} \mathbf{i}_1 + w_{T_2} \mathbf{i}_2 + w_{T_3} \mathbf{i}_3) \circ \mathbf{B}(\theta) &= 0. \end{aligned} \quad (7)$$

Далее была получена соответствующая (5), сопряженная система:

$$\begin{cases} 2\dot{\boldsymbol{\Psi}} = \boldsymbol{\Psi} \circ \boldsymbol{\omega}, \\ \dot{\rho} = 0, \\ \dot{\boldsymbol{\varphi}} = -\text{vect}(\tilde{\mathbf{\Lambda}} \circ \boldsymbol{\Psi})/2 - b_2 \rho \mathbf{i}_1. \end{cases} \quad (8)$$

Для решения поставленной задачи были использованы следующие библиотеки *Python*:

1. *Sympy* - система компьютерной алгебры, включающая в себя функции базовой символьной арифметики, математического анализа, алгебры и дискретной математики, элементов квантовой физики, с возможностью форматирования результата в код \LaTeX .
2. *Scipy* - библиотека, предназначенная для выполнения научных и инженерных вычислений.
3. *Matplotlib* - библиотека для визуализации данных.

Сначала в файле *main.py* задаются моменты инерции в размерных величинах, затем начальные и граничные условия L_0, L_T, w_0, w_T в безразмерных переменных:

Далее с помощью метода *symbolic* из файла *symbolic.py* определяется вид формул для дальнейшего расчета.

На вход функции подаются безразмерные моменты инерции, вектор угловой скорости КА в конечный момент времени, константы b_1, b и b_2 .

```
1 def symbolic(Ib, wT, b_ch, b1_ch, b2_ch):
```

Далее объявляются переменные x, ψ, u :

1. x — массив переменных основной системы.
2. psi — массив переменных сопряженной системы.
3. u — параметры управления.

4. b_1, b, b_2 — константы.

5. I_1, I_2 — моменты инерции осесимметричного тела.

Затем моменты инерции сводятся к безразмерным величинам с помощью коэффициента масштабирования:

$$I^{\text{масш}} = ((I_1^2 + 2I_2^2) / 3)^{1/2}, \quad I_k^{\text{безраз}} = I_k / I^{\text{масш}}, \quad k = 1, 2;$$

```
1 I_mash = ((I[1] ** 2 + 2 * I[2] ** 2) / 3) ** (1 / 2)
2 Ib = [I[i] / I_mash for i in [1, 2]]

1     u = sp.symbols('u_1:4')
2     x = sp.symbols('x_1:9')
3     psi = sp.symbols('psi_0:9')
4     b1, b, b2 = sp.symbols('b_1 b b_2')
```

С помощью метода *Quaternion* из пакета *Sympy* объявляются, в кватернионной постановке, переменные основной и сопряженной системы системы: Λ, Ψ, ω .

```
1     lambd = Quaternion(*x[0:4])
2     Psi = Quaternion(*psi[1:5])
3     omega = Quaternion(0, *x[4:7])
```

Создается список, в котором содержатся правые части уравнений дифференциальной системы.

```
1 fi = [-cost(u, Ib), *Lambd.args, *u, b2 * x[4]]
```

Составляется функция Гамильтона-Понтрягина.

```
1     H = Hamilton(psi, fi)
```

Далее полученные формулы преобразуются из символьных вычислений в функции *Python*, принимающие на вход переменные основной и сопряженной систем.

```
1     Hf = sp.lambdify((*x, *psi[1:]), H)
2     fs = [*f_x, *f_psi]
3     fs = [sp.lambdify((*x, *psi[1:]), i, 'scipy') for i in fs]
```

Далее рассчитываются численные значения в методе *solve* из файла *scr.py*

```
1 data = solve(L0, LT, w0, b1_ch, *data, printT=True)
```

root - поиск минимума вектор функции.

odeint - решение дифференциальных уравнений первого порядка.

На вход функции невязок передаётся начальное приближение начальных, значений сопряжённой системы. Затем создается линейное пространство T на котором решается система дифференциальных уравнений.

Сначала задается начальное приближение сопряженных координат:

```
1 psi0 = [1, 0, -1, 0, 0, -1, 0, 1, 1]
```

Затем составляется функция F , возвращающая вектор невязок

```
1 def koshi(y0, t):
2     return [i(*y0) for i in fs]
3
4 ##### Нелинейное уравнение #####
5     def F(psi0):
6
7         T = psi0[0]
8
9         tn = np.linspace(0, T)
10
11        Lambda0 = L0_to_Lambda0(L0)
12        omega0 = w0_to_omega0(w0, b1_ch)
13        theta0 = 0
14
15        y0 = [*Lambda0.args, *omega0.args[1:], theta0, *psi0[1:]]
16        yT = odeint(koshi, y0, tn)[-1]
17        thetaT = yT[7]
18
19        LambdaT = Quaternion(*yT[0:4])
20        omegaT = Quaternion(0, *yT[4:7])
21        PsiT = Quaternion(*yT[8:12])
22        phiT = Quaternion(0, *yT[12:15])
23        ro0 = yT[15]
24        p = vector_part(LambdaT.conjugate() * PsiT)
25        return [Hf(*yT),
```

```

26     transversal(*yT),
27     *[i(*yT) for i in omega_boundary_f],
28     *(-LambdaT * B(thetaT).conjugate() * LT.conjugate()).args[1:],
29     ro0 - p.b / 2 + phiT.c * omegaT.d - phiT.d * omegaT.c]

```

Полученные численные значения в конечный момент времени должны удовлетворять следующим равенствам:

$$\begin{aligned}
H(T) &= -\psi^* (1 + I_2^2 \mathbf{u}^2) + (\Psi, \Lambda \circ \omega) / 2 + (\varphi, \mathbf{u}) + b_2 \omega_1 \rho = 0, \\
\text{vect} \left(\Lambda \circ \tilde{\mathbf{B}}(\theta) \circ \tilde{\mathbf{L}}_T \right) &= 0, \\
\omega - \tilde{\mathbf{B}}(\theta) \circ (b_1 w_{T_1} \mathbf{i}_1 + w_{T_2} \mathbf{i}_2 + w_{T_3} \mathbf{i}_3) \circ \mathbf{B}(\theta) &= 0, \\
\text{scal}(\tilde{\Lambda} \circ \Psi(t)) &= 0,
\end{aligned}$$

где «scal (.)» обозначает скалярную часть кватерниона, и

$$\rho_0 + p_1(T)/2 + \varphi_2(T)\omega_3(T) - \varphi_3(T)\omega_2(T) = 0.$$

Это и является вектором невязок, нули которого ищутся с помощью метода root.

```

1     sol = root(F, psi0)

```

Наконец, полученные значения визуализируют посредством функции show_graph из файла *show_graph.py*

```

1     show_graph(*data)

```

В ходе работы были проанализированы различные численные методы решения дифференциальных уравнений. Наилучшую скорость работы показал метод Пауэля или гибридный метод.

В бакалаврской работе были рассмотрены варианты решения задачи для МКС и «Space Shuttle», в автореферате приведён пример для МКС.

Пример для МКС

$$L_0 = 0.79505 + 0.29814i - 0.39752j + 0.34783k,$$

$$L_T = 1 + 0i + 0j + 0k,$$

$$w_0 = 0 + 0i + 0j + 0k,$$

$$w_T = 0 + 0i + 0j + 0k,$$

$$I1 = 4851826875 \text{ кг} \cdot \text{м}^2,$$

$$I2 = 2377395168.75 \text{ кг} \cdot \text{м}^2.$$

Проанализируем как изменяются: время, вектор угловой скорости, вектор управления и значения функционала при изменении параметра α .

По заданным граничным условиям и величине моментов инерции получим следующие графики изменения компонент кватерниона ориентации.

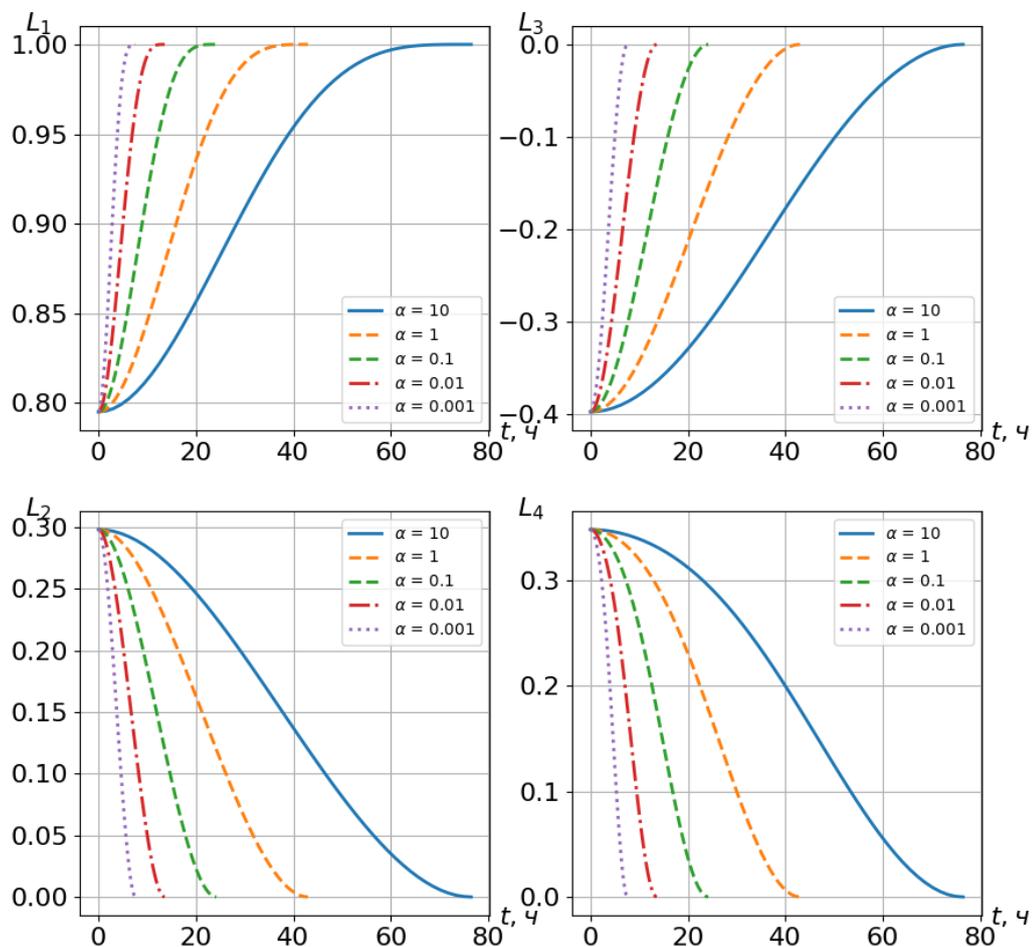


Рисунок 1 — График изменения кватерниона ориентации \mathbb{L}

Принимая параметр $\alpha = 1$ за основной, будем фиксировать изменения времени, вектора угловой скорости, вектора управления и значения функционала относительно него. В соответствии с таблицей 1 заметим следующую

Таблица 1 — Зависимость переменных системы от параметра α

	$M_1, \frac{\text{Н}}{\text{М}}$	$M_2, \frac{\text{Н}}{\text{М}}$	$M_3, \frac{\text{Н}}{\text{М}}$	$W_1, \frac{\text{рад}}{\text{ч}}$	$W_2, \frac{\text{рад}}{\text{ч}}$	$W_3, \frac{\text{рад}}{\text{ч}}$	$J \cdot 10^6, \text{с}$	$T, \text{ч}$
$\alpha = 10$	0.159	0.136	0.125	0.011	0.018	0.016	0.371	76.450
$\alpha = 1$	0.504	0.429	0.395	0.020	0.032	0.028	0.208	42.991
$\alpha = 0.1$	1.595	1.358	1.248	0.036	0.056	0.050	0.117	24.175
$\alpha = 0.01$	5.044	4.294	3.946	0.065	0.100	0.089	0.066	13.595
$\alpha = 0.001$	15.949	13.579	12.478	0.115	0.178	0.157	0.037	7.645

закономерность: при увеличении параметра α увеличивается время и значение функционала, а компоненты вектора управления и вектора поворота уменьшаются.

По полученным результатам в соответствии с таблицей 1 и в соответствии с рисунком 1 имеем, что параметр α прямо-пропорционально масштабирует результат исходной безразмерной задачи.

Заключение

В данной работе было проведено исследование и анализ задачи разворота КА с использованием символьной алгебры. Была изучена модель приведения космического аппарата в нужное состояние. Было получено численное решение данной задачи с использованием символьной алгебры и рассмотрены примеры для различных существующих космических аппаратов.