

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное автономное образовательное
учреждение высшего образования
«СЕВАСТОПОЛЬСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

ИНСТИТУТ РАДИОЭЛЕКТРОНИКИ
И ИНТЕЛЛЕКТУАЛЬНЫХ ТЕХНИЧЕСКИХ СИСТЕМ

Кафедра "Информатика и управление в технических системах"

ЛИНЕАРИЗАЦИЯ МОДЕЛЕЙ ДИНАМИЧЕСКИХ УПРАВЛЯЕМЫХ ОБЪЕКТОВ

Методические указания
к выполнению лабораторной работы
по дисциплине "Основы управления в технических системах"
для студентов направления подготовки
27.03.04 – Управление в технических системах



Севастополь
СевГУ
2025

УДК 681.511.48(076.5)
ББК 32.965.4я73
Л591

Р е ц е н з е н т:

А. А. Кабанов – канд. техн. наук, доцент,
заведующий кафедрой "Информатика и управление в технических системах"

Составители: В. А. Карапетьян, В. А. Крамарь

Л591 Линеаризация моделей динамических управляемых объектов : методические указания к выполнению лабораторной работы по дисциплине "Основы управления в технических системах" для студентов направления подготовки 27.03.04 – Управление в технических системах / Севастопольский государственный университет, Институт радиоэлектроники и интеллектуальных технических систем, кафедра «Информатика и управление в технических системах» ; сост. В. А. Карапетьян, В. А. Крамарь. – Севастополь : СевГУ, 2025. – 23 с. – Текст : электронный.

Приводится описание лабораторной работы, целью которой является исследование процедур линеаризации, построение моделей стационарных линейных динамических управляемых объектов, применение средств среды Matlab&Simulink для решения таких задач.

Предназначены для студентов дневной и заочной форм обучения направления подготовки 27.03.04 – Управление в технических системах.

УДК 681.511.48(076.5)
ББК 32.965.4я73

Рассмотрены и рекомендованы к изданию на заседании кафедры "Информатика и управление в технических системах", протокол № 5 от 28 января 2025 г.

© Карапетьян В. А., Крамарь В. А., сост., 2025
© ФГАОУ ВО «Севастопольский
государственный университет», 2025

СОДЕРЖАНИЕ

1 ЦЕЛЬ РАБОТЫ.....	4
2 МАТЕМАТИЧЕСКИЕ МОДЕЛИ ЛИНЕЙНЫХ УПРАВЛЯЕМЫХ ОБЪЕКТОВ И СИСТЕМ.....	4
2.1 Техника линеаризации	5
2.2 Формы математических моделей ЛСС.....	7
3 ПРОГРАММА И ПОРЯДОК ВЫПОЛНЕНИЯ РАБОТЫ	8
4 НЕОБХОДИМЫЕ СВЕДЕНИЯ ДЛЯ ВЫПОЛНЕНИЯ ЛАБОРАТОРНОЙ РАБОТЫ.....	9
5 СОДЕРЖАНИЕ ОТЧЁТА И ПОРЯДОК ЗАЩИТЫ РАБОТЫ	11
6 КОНТРОЛЬНЫЕ ВОПРОСЫ.....	12
БИБЛИОГРАФИЧЕСКИЙ СПИСОК	13
ПРИЛОЖЕНИЕ А Пример решения основных задач лабораторной работы.....	14

1 ЦЕЛЬ РАБОТЫ

Приобретение умений и практических навыков применения методов и процедур линеаризации математических моделей динамических управляемых объектов, построения различных форм моделей линейных стационарных систем управления и моделирование их движения. Формирование навыков профессиональной работы в системе Matlab&Simulink.

2 МАТЕМАТИЧЕСКИЕ МОДЕЛИ ЛИНЕЙНЫХ УПРАВЛЯЕМЫХ ОБЪЕКТОВ И СИСТЕМ

В лабораторной работе рассматриваются процедуры линеаризации математических моделей динамических управляемых объектов (систем нелинейных обыкновенных дифференциальных уравнений (ОДУ)). Линеаризация проводится в окрестности заданной рабочей точки. В результате проведения процедуры линеаризации получают математическую модель движения управляемого объекта в малых отклонениях от точки линеаризации. Такие модели представляются системами линейных ОДУ относительно переменных, описывающих эти малые отклонения [1, 2, 3].

Автоматические системы стабилизации и слежения практически всегда проектируются в контексте управления объектом в малой окрестности номинального режима его работы. К тому же, для таких систем, условия и временные интервалы функционирования часто позволяют считать их параметры постоянными. Методы анализа и проектирования линейных стационарных систем управления разработаны наиболее полно и составляют фундаментальные основы ТАУ [1-5].

В англоязычной литературе линейные динамические системы с постоянными коэффициентами обозначают как LTI (Linear Time-Invariant) системы. В русскоязычной литературе такие системы принято называть линейными стационарными системами – ЛСС. В настоящее время установилась терминология и для обозначения одномерных и многомерных систем. Одномерные системы – системы, имеющие один вход и один выход – SISO-системы – Single-Input/Single-Output. Многомерные системы – системы, имеющие несколько входов и несколько выходов – MIMO-системы – Multiple-Input/Multiple-Output [4, 5].

Зададим математическую модель движения управляемого объекта в терминах "вход-состояние-выход" [3, 4]:

$$\begin{cases} \dot{x}(t) = f[x(t), u(t), t], \\ y(t) = g[x(t), u(t), t]. \end{cases} \quad (2.1)$$

$$(2.2)$$

Здесь уравнение (2.1) – *уравнение состояния* – система нелинейных ОДУ в нормальной форме; выражение (2.2) – *уравнение выходов* – система нелинейных алгебраических функциональных соотношений.

Вектором *состояния* $\mathbf{x}(t)$ модели обозначают совокупность внутренних переменных системы, значения которых наряду со знанием входов $\mathbf{u}(t)$ и уравнений математической модели, позволяют определить дальнейшее поведение

(движение) системы $\mathbf{y}(t)$. В качестве вектора входов $\mathbf{u}(t)$ выступают переменные, описывающие внешние причины возникновения движения. Вектор выходов $\mathbf{y}(t)$ – это управляемые переменные системы, значения которых, обычно, доступны наблюдению (измерению). Размерности векторов-функций состояния $\mathbf{x}(t)$, входов $\mathbf{u}(t)$ и выходов $\mathbf{y}(t)$ будем обозначать соответственно n , m и p .

2.1 Техника линеаризации

В математическом смысле, *линеаризация* – это аппроксимация (приближённая замена) нелинейных зависимостей (уравнений) менее точными, но более простыми – линейными зависимостями (уравнениями). В нашем случае линеаризации подвергаются нелинейные уравнения (2.1) и (2.2), представляющие математическую модель движения (динамики) управляемой системы.

Процедура линеаризации уравнений модели состоит в следующем:

– пусть управляемая система, описываемая моделью (2.1) и (2.2), движется (функционирует, работает) по номинальной (опорной) траектории $\mathbf{x}_0(t)$ под действием заданного известного входного управляющего воздействия $\mathbf{u}_0(t)$. В этом случае система уравнений модели также справедлива для математического описания и этого движения $[\mathbf{x}_0(t), \mathbf{y}_0(t), \mathbf{u}_0(t)]$, т.е.,

$$\begin{cases} \dot{\mathbf{x}}_0(t) = \mathbf{f}[\mathbf{x}_0(t), \mathbf{u}_0(t), t], \\ \mathbf{y}_0(t) = \mathbf{g}[\mathbf{x}_0(t), \mathbf{u}_0(t), t]; \end{cases} \quad (2.3)$$

– введём "возмущённые" вектор-функции (переменные) $\mathbf{x}(t) = \mathbf{x}_0(t) + \Delta\mathbf{x}(t)$, $\mathbf{y}(t) = \mathbf{y}_0(t) + \Delta\mathbf{y}(t)$ и $\mathbf{u}(t) = \mathbf{u}_0(t) + \Delta\mathbf{u}(t)$, где $\Delta\mathbf{x}(t)$, $\Delta\mathbf{y}(t)$ и $\Delta\mathbf{u}(t)$ – "малые" отклонения от номинальной траектории $[\mathbf{x}_0(t), \mathbf{y}_0(t), \mathbf{u}_0(t)]$ по переменным состояния $\Delta\mathbf{x}(t)$ и выхода $\Delta\mathbf{y}(t)$, вызванные малыми отклонениями вектор-функции входной переменной $\Delta\mathbf{u}(t)$. Система уравнений (2.1), (2.2) справедлива и для этого "возмущённого", относительного номинального, движения –

$$\begin{cases} (\mathbf{x}_0(t) + \Delta\mathbf{x}(t))' = \mathbf{f}[\mathbf{x}_0(t) + \Delta\mathbf{x}(t), \mathbf{u}_0(t) + \Delta\mathbf{u}(t), t], \\ \mathbf{y}_0(t) + \Delta\mathbf{y}(t) = \mathbf{g}[\mathbf{x}_0(t) + \Delta\mathbf{x}(t), \mathbf{u}_0(t) + \Delta\mathbf{u}(t), t]; \end{cases} \quad (2.4)$$

– предполагая достаточную гладкость функций $\mathbf{f}$ и $\mathbf{g}$ по своим аргументам, представим вектор-функции правых частей $\mathbf{f}(\cdot)$ и $\mathbf{g}(\cdot)$ уравнений системы (2.4) в виде рядов Тейлора в окрестности номинального режима $[\mathbf{x}_0(t), \mathbf{y}_0(t), \mathbf{u}_0(t)]$. Так как функции правых частей $\mathbf{f}(\cdot)$ и $\mathbf{g}(\cdot)$, являясь векторными, сами зависят от векторных аргументов $\mathbf{x}$ и $\mathbf{u}$, коэффициенты ряда Тейлора будут представлять собой матричные функции, определяемые текущими значениями номинального процесса. В силу малости текущих отклонений $[\Delta\mathbf{x}(t), \Delta\mathbf{y}(t), \Delta\mathbf{u}(t)]$ ограничимся только линейными, по этим отклонениям, членами ряда Тейлора. В итоге – получим линейную нестационарную нормальную форму системы ОДУ и линейную нестационарную алгебраическую функциональную систему уравнений относительно отклонений $[\Delta\mathbf{x}(t), \Delta\mathbf{y}(t), \Delta\mathbf{u}(t)]$:

$$\begin{cases} \frac{d}{dt} \Delta x(t) = A(t) \cdot \Delta x(t) + B(t) \cdot \Delta u(t), \\ \Delta y(t) = C(t) \cdot \Delta x(t) + D(t) \cdot \Delta u(t). \end{cases} \quad (2.5)$$

Здесь $A(t)$, $B(t)$, $C(t)$, $D(t)$ – матричные функции Якоби (якобианы) соответствующих размерностей, в общем случае, зависящие от времени:

$$\begin{aligned} A[x_0(t), u_0(t), t] &= \frac{\partial f}{\partial x} \Big|_{x_0(t), u_0(t)}, & B[x_0(t), u_0(t), t] &= \frac{\partial f}{\partial u} \Big|_{x_0(t), u_0(t)}, \\ C[x_0(t), u_0(t), t] &= \frac{\partial g}{\partial x} \Big|_{x_0(t), u_0(t)}, & D[x_0(t), u_0(t), t] &= \frac{\partial g}{\partial u} \Big|_{x_0(t), u_0(t)}, \end{aligned} \quad (2.6)$$

где все частные производные вычисляются вдоль номинальной траектории движения объекта управления $[x_0(t), u_0(t)]$.

В скалярной форме матрицы Якоби (2.6) модели (2.5) имеют вид

$$\begin{aligned} A(t) = \frac{\partial f}{\partial x^T} \Big|_{x_0, u_0} &= \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \frac{\partial f_1}{\partial x_2} & \dots & \frac{\partial f_1}{\partial x_n} \\ \frac{\partial f_2}{\partial x_1} & \frac{\partial f_2}{\partial x_2} & \dots & \frac{\partial f_2}{\partial x_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial f_n}{\partial x_1} & \frac{\partial f_n}{\partial x_2} & \dots & \frac{\partial f_n}{\partial x_n} \end{bmatrix}; & B(t) = \frac{\partial f}{\partial u^T} \Big|_{x_0, u_0} &= \begin{bmatrix} \frac{\partial f_1}{\partial u_1} & \frac{\partial f_1}{\partial u_2} & \dots & \frac{\partial f_1}{\partial u_m} \\ \frac{\partial f_2}{\partial u_1} & \frac{\partial f_2}{\partial u_2} & \dots & \frac{\partial f_2}{\partial u_m} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial f_n}{\partial u_1} & \frac{\partial f_n}{\partial u_2} & \dots & \frac{\partial f_n}{\partial u_m} \end{bmatrix}; \\ C(t) = \frac{\partial g}{\partial x^T} \Big|_{x_0, u_0} &= \begin{bmatrix} \frac{\partial g_1}{\partial x_1} & \frac{\partial g_1}{\partial x_2} & \dots & \frac{\partial g_1}{\partial x_n} \\ \frac{\partial g_2}{\partial x_1} & \frac{\partial g_2}{\partial x_2} & \dots & \frac{\partial g_2}{\partial x_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial g_p}{\partial x_1} & \frac{\partial g_p}{\partial x_2} & \dots & \frac{\partial g_p}{\partial x_n} \end{bmatrix}; & D(t) = \frac{\partial g}{\partial u^T} \Big|_{x_0, u_0} &= \begin{bmatrix} \frac{\partial g_1}{\partial u_1} & \frac{\partial g_1}{\partial u_2} & \dots & \frac{\partial g_1}{\partial u_m} \\ \frac{\partial g_2}{\partial u_1} & \frac{\partial g_2}{\partial u_2} & \dots & \frac{\partial g_2}{\partial u_m} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial g_p}{\partial u_1} & \frac{\partial g_p}{\partial u_2} & \dots & \frac{\partial g_p}{\partial u_m} \end{bmatrix}. \end{aligned}$$

Если матричные коэффициенты уравнений (2.5) не зависят от времени, то линеаризованная модель представляется системой из n линейных стационарных ОДУ в нормальной форме и из p линейных алгебраических функциональных соотношений. Для упрощения записи обозначение приращения в виде символа D опускают и систему уравнений (2.5) записывают в виде

$$\begin{cases} x'(t) = A \cdot x(t) + B \cdot u(t), \\ y(t) = C \cdot x(t) + D \cdot u(t). \end{cases} \quad (2.7)$$

Соотношения (2.7) представляют собой математическую модель линейной стационарной управляемой динамической системы в форме "вход-состояние-выход" – модель пространства состояний линейной стацио-

нарной системы (ЛСС или ЛТИ) n -го порядка с m входами и p выходами. Числовые матрицы коэффициентов имеют специальные наименования:

- $n \times n$ матрица A – матрица динамических коэффициентов системы;
- $n \times m$ матрица B – матрица входов системы;
- $p \times n$ матрица C – матрица выходов системы;
- $p \times m$ матрица D – матрица прямой связи входа с выходом системы.

2.2 Формы математических моделей ЛСС

Математические модели линейных стационарных систем управления обычно представляют в одной из двух форм:

- в форме моделей временной области – временные модели:
 - модель "вход-выход"

$$\begin{aligned} a_n y^{(n)}(t) + a_{n-1} y^{(n-1)}(t) + \dots + a_1 y'(t) + a_0 y(t) = \\ = b_m u^{(m)}(t) + b_{m-1} u^{(m-1)}(t) + \dots + b_1 u'(t) + b_0 u(t); \end{aligned} \quad (2.8)$$

- модель "вход-состояние-выход" – система уравнений (2.7)

$$\begin{cases} x'(t) = A \cdot x(t) + B \cdot u(t), \\ y(t) = C \cdot x(t) + D \cdot u(t); \end{cases}$$

- в форме моделей комплексной (частотной) области:

– модель "вход-выход" в виде передаточной функции – ПФ (в англоязычной литературе – transfer function). Передаточная функция связывает собою изображения по Лапласу входа $U(s)$ и выхода $Y(s)$ динамической системы. Для случая SISO-системы, т.е. при $m=1$ и $p=1$, имеем:

$$\begin{aligned} Y(s) = \frac{b_m s^m + b_{m-1} s^{m-1} + \dots + b_1 s + b_0}{a_n s^n + a_{n-1} s^{n-1} + \dots + a_1 s + a_0} U(s) \Rightarrow Y(s) = W(s) U(s), \\ W(s) = \frac{Y(s)}{U(s)} = \frac{b_m s^m + b_{m-1} s^{m-1} + \dots + b_1 s + b_0}{a_n s^n + a_{n-1} s^{n-1} + \dots + a_1 s + a_0} = \frac{P(s)}{Q(s)}. \end{aligned} \quad (2.9)$$

Здесь $s = \sigma + j\omega$ – комплексная переменная преобразования Лапласа; $Y(s)$ и $U(s)$ – изображения по Лапласу входа $u(t)$ и выхода $y(t)$ системы;

$W(s)$ – передаточная функция системы;

$P(s)$ и $Q(s)$ – полиномы (многочлены) числителя и знаменателя ПФ по переменной Лапласа s , соответственно m -й и n -й степеней.

В соответствии с выражениями (2.7), ОДУ (2.8) и свойствами преобразования Лапласа, можно дать ещё одно определение передаточной функции SISO-системы:

Передаточной функцией системы называют дробно-рациональную функцию, являющуюся несократимым отношением изображений по Лапласу выходного сигнала к входному сигналу при нулевых начальных условиях в системе.

Полином знаменателя ПФ $Q(s)$ называют характеристическим полиномом (многочленом) системы, а его степень n – порядком системы;

– модель "вход-состояние-выход" в виде матричной передаточной функции. Матричная передаточная функция (МПФ) связывает собою изображения по Лапласу вектора входа $U(s)$ и вектора выхода $Y(s)$ ЛСС. Выражение для МПФ можно легко получить, применив преобразование Лапласа к уравнениям модели (2.7), положив начальные условия для вектора состояния нулевыми, т.е., $x(0)=0$:

$$\begin{aligned} sX(s) &= A \cdot X(s) + B \cdot U(s) \Rightarrow (s \cdot I - A) \cdot X(s) = B \cdot U(s) \Rightarrow \\ X(s) &= (s \cdot I - A)^{-1} B \cdot U(s) \Rightarrow Y(s) = C \cdot X(s) + B \cdot U(s) \Rightarrow \\ Y(s) &= [C(sI - A)^{-1} B + D] \cdot U(s) \Rightarrow Y(s) = W(s)U(s) \Rightarrow \\ W(s) &= C(sI - A)^{-1} B + D. \end{aligned} \quad (2.10)$$

Здесь $W(s)$ – матричная передаточная функция размерности $p \times m$. Элементами МПФ $W(s)$ являются дробно-рациональные функции; I – единичная матрица размерности n .

В большинстве практических случаев в динамических объектах прямое влияние входных воздействий $u(t)$ на выход системы $y(t)$ отсутствует, поэтому в линеаризованных моделях таких систем матрица прямых связей D оказывается нулевой. Таким образом, для большинства случаев МПФ будет представлена следующим выражением:

$$W(s) = C(sI - A)^{-1} B = \frac{1}{\det(sI - A)} C \cdot \text{adj}(sI - A) \cdot B. \quad (2.11)$$

Здесь $\det(sI - A) = q(s)$ – характеристический полином матрицы A . Порядок полинома – n ;

$\text{adj}(sI - A)$ – присоединённая матрица для матрицы $(sI - A)$;

$(sI - A)^{-1}$ – матрица резольвента матрицы A .

3 ПРОГРАММА И ПОРЯДОК ВЫПОЛНЕНИЯ РАБОТЫ

В работе выполняются процедуры линеаризации математических моделей управляемых динамических объектов в малой окрестности заданных номинальных режимов (рабочих точек). В результате линеаризации должны быть получены модели ЛСС во временной и комплексной областях. В качестве объекта линеаризации выбирается одна из моделей из [6]. Точкой линеаризации следует выбрать положение устойчивого равновесия динамического объекта.

В лабораторной работе следует исполнить следующие задания:

- провести линеаризацию уравнений модели в тремя различными способами:
 - 1) аналитически вручную;

- 2) аналитически с использованием средств среды Matlab и библиотеки функций символьных вычислений Symbolic Math Toolbox [7, 8];
- 3) с использованием средств среды Matlab [9] и пакета Simulink [10]:
 - функция **linmod**;
 - инструментальный приложения Model Linearizer [11];
- полученные линеаризованные модели ЛСС представить во временной и комплексной областях – уравнениями пространства состояний и передаточными функциями;
- построить графики собственного движения ЛСС с помощью:
 - функций-решателей систем ОДУ среды Matlab [9];
 - функция **initial** библиотеки Control System Toolbox [12];
 - средств моделирования ЛСС в Simulink;
 - по символьным выражениям для переменных состояния, используя средства библиотеки Symbolic Math Toolbox [7, 8];
- выполнить моделирование собственного движения объекта по его *нелинейной модели* [6], построить соответствующие графики и сравнить их с результатами моделирования ЛСС, полученными в предыдущем пункте. Начальные условия для моделирования собственного движения выбрать близкими к значениям координат точки линеаризации;
- провести сравнительный анализ результатов ручных и программных расчётов и графических построений;
- сформулировать общие выводы по работе.

4 НЕОБХОДИМЫЕ СВЕДЕНИЯ ДЛЯ ВЫПОЛНЕНИЯ ЛАБОРАТОРНОЙ РАБОТЫ

Выполнение задачи лабораторной работы предполагает использование функций ядра Matlab, специализированных инструментальных библиотек Control System Toolbox и Symbolic Math Toolbox, средств системы Simulink.

Для выполнения собственно процедуры линеаризации средствами среды Matlab и пакета символьных вычислений Symbolic Math Toolbox предлагается использовать функцию **jacobian**. Функция **jacobian** из пакета Symbolic Math Toolbox вычисляет матрицу Якоби от вектора-функции **f** по вектору **v**:

$$J = \frac{\partial f}{\partial v^T} = \left\{ J_{i,j} = \frac{\partial f_i}{\partial v_j} \middle| \begin{matrix} i=\overline{1,m} \\ j=\overline{1,n} \end{matrix} \right\} = \begin{bmatrix} \frac{\partial f_1}{\partial v_1} & \frac{\partial f_1}{\partial v_2} & \dots & \frac{\partial f_1}{\partial v_n} \\ \frac{\partial f_2}{\partial v_1} & \frac{\partial f_2}{\partial v_2} & \dots & \frac{\partial f_2}{\partial v_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial f_m}{\partial v_1} & \frac{\partial f_m}{\partial v_2} & \dots & \frac{\partial f_m}{\partial v_n} \end{bmatrix}.$$

Здесь **f** и **v** – символьные векторы размерностей **m** и **n** соответственно.

После получения символьного выражения для матрицы Якоби **J** следует определить численное значение матрицы **J**. Для этого используется функция **subs**, предназначенная для подстановки численных значений символьных переменных в символьное выражение.

Линеаризация с использованием пакета Simulink и функции **linmod** позволяет получать модели ЛСС как в форме пространства состояний, так и в форме передаточных функций, основываясь на Simulink-схеме, построенной в соответствии с исходной нелинейной моделью в нормальной форме. Построение необходимой здесь Simulink-схемы проведено в [6]. Количество входов и количество выходов нелинейной математической модели функция **linmod** извлекает из Simulink-схемы, определяя в ней наличие блоков Inport и Outport.

Функция **linmod** может быть вызвана, например, следующий образом:

```
[A,B,C,D] = linmod('sys', x, u);
```

Здесь **A**, **B**, **C**, **D** – возвращаемые матрицы модели пространства состояний линеаризованной ЛСС; **sys** – имя Simulink-файла с Simulink-моделью исходной нелинейной системы; **x** – числовой вектор состояния, содержащий координаты точки в пространстве состояний системы, в окрестности которой проводится линеаризация; **u** – числовой вектор входов, содержащий значения постоянных входных воздействий, соответствующих рабочей точки линеаризации.

Если синтаксис вызова функции **linmod** предлагает два возвращаемых параметра, то первый из них будет содержать коэффициенты числителя ПФ, а второй – коэффициенты знаменателя ПФ (характеристического полинома системы) ЛСС.

Средство Model Linearizer предназначено для проведения процедур линеаризации как отдельных Simulink-блоков, так и сложных Simulink-схем нелинейных динамических систем. Инструментарий применяется к готовой Simulink-схеме через главное меню Simulink-редактора APPS > Model Linearizer. Параметры блоков Simulink-схемы определяются значениями координат точки линеаризации (значения компонент вектора состояния динамической системы и соответствующие величины входных воздействий). Линеаризованная система возвращается инструментом Model Linearizer в виде **ss**-, **zpk**- или **tf**-модели.

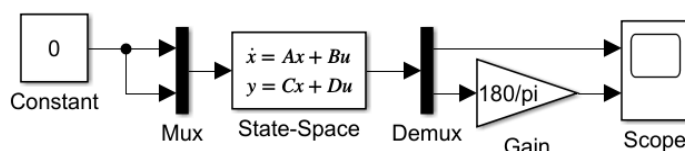
Формирование математических моделей ЛСС в среде Matlab производится с помощью нескольких специальных функций из пакета Control System Toolbox [12] – для решения задач текущей лабораторной работы потребуются две из них – **tf** и **ss**:

– функция **tf** (*transfer function*) создаёт линейный стационарный динамический объект (ЛСС или LTI-объект) в виде ПФ **W** по известным коэффициентам полиномов числителя **num** и знаменателя **den** ПФ. Соответствующая инструкция имеет вид: **W=tf(num,den)**, где **num** и **den** – одномерные массивы коэффициентов полиномов. Коэффициенты упакованы в массивы в порядке убывания степеней переменной полинома;

– функция **ss** (*state space*) создаёт линейный стационарный динамический объект **sys** (ЛСС или LTI-объект) в виде модели пространства состояний по известным матричным коэффициентам **A**, **B**, **C**, **D**. Соответствующая инструкция имеет вид: **sys=ss(A,B,C,D)**, где **A**, **B**, **C**, **D** – матрицы коэффициентов размерностей $n \times n$, $n \times m$, $p \times n$ и $p \times m$.

Для построения графиков собственного движения ЛСС следует использовать функцию **initial** из пакета Control System Toolbox. Функция работает исключительно с моделями ЛСС в пространстве состояний. Наиболее простой способ решения данной задачи состоит в вызове этой функции в таком формате: **initial(sys,x0)**. Здесь **sys** – LTI-объект, заданный в терминах пространства состояний; **x0** – вектор начального состояния ЛСС.

Решение аналогичной задачи в Simulink состоит в моделировании собственного движения системы и построении графиков с помощью, например, следующей Simulink-схемы:



где блок State-Space представляет LTI-объект в пространстве состояний;

блок Scope – имитирует здесь двухэкранный осциллограф;

блок Constant – генерирует постоянный (в данном случае, нулевой) внешний скалярный сигнал;

блок Mux – концентратор – объединяет несколько скалярных сигналов в один векторный;

блок Demux – извлекает компоненты из входного векторного сигнала и выводит их в отдельные сигналы.

В данном примере наличие блоков Mux и Demux вызвано необходимостью организации векторных входа $u(t)$ и выхода $y(t)$, т.к. предполагаемый здесь объект имеет два входа и два выхода ($m=p=2$); блок Gain – усилитель – в данном случае выступает преобразователем радианной меры угла в градусную.

Построение графика собственного движения по найденному аналитическому выражению следует осуществлять функцией **fplot** из библиотеки Symbolic Math Toolbox.

5 СОДЕРЖАНИЕ ОТЧЁТА И ПОРЯДОК ЗАЩИТЫ РАБОТЫ

В отчёте по лабораторной работе должны быть отражены цель, предварительные математические построения и расчёты, необходимые сведения о методах решения рассматриваемых задач и средствах системы Matlab, тексты программ, схемы моделирования, численные и графические результаты расчётов и построений, выводы по работе.

Защита результатов лабораторной работы производится за компьютером при наличии работающих программ и полностью оформленного отчёта. В ходе

защиты студент может получить дополнительные теоретические вопросы и практические задания по существу работы.

6 КОНТРОЛЬНЫЕ ВОПРОСЫ

1. Поясните смысл процедуры линеаризации.
2. Обоснуйте идею линеаризации аналитически и графически.
3. Поясните этапы процедуры линеаризации модели в пространстве состояний – линеаризация системы нелинейных ОДУ в нормальной форме?
4. Какие Вы знаете формы представления математических моделей ЛСС во временной области?
5. Какие Вы знаете формы представления математических моделей ЛСС в комплексной области?
6. Какие свойства преобразования Лапласа используются при переходе от моделей временной области к моделям комплексной области?
7. Поясните применение функции **jacobian**.
8. Почему функция **initial** воспринимает LTI-объект только в форме модели пространства состояний?
9. Каким блокам Simulink-схемы сопоставлены значения координат рабочей точки из второго параметра функции **linmod**?
10. Поясните порядок действий с инструментом Model Linearizer.
11. Каким образом можно передать полученную в утилите Model Linearizer линейную модель ЛСС в рабочую область Matlab.

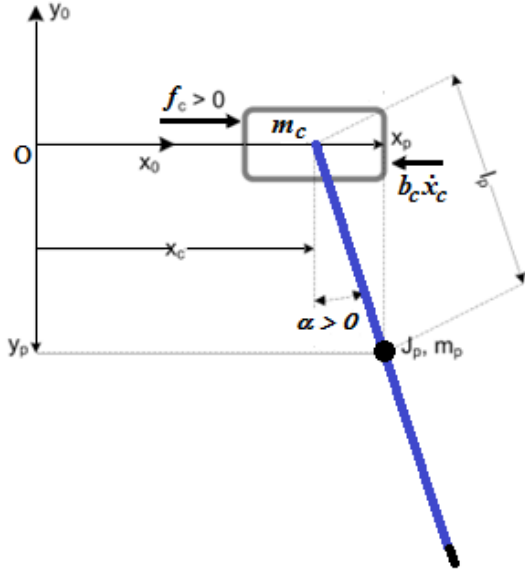
БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Математические основы теории автоматического управления. В 3 т. Т.1. : учеб. пособие / В. А. Иванов, В. С. Медведев, Б. К. Чемоданов, А. С. Ющенко ; под ред. Б. К. Чемоданова. – 3-е изд., перераб. и доп. — М. : МГТУ им. Н. Э. Баумана, 2006. — 552 с. — Текст : непосредственный.
2. Ким, Д. П. Теория автоматического управления. Линейные системы : учебник и практикум для вузов / Д. П. Ким. — 3-е изд., испр. и доп. — Москва : Юрайт, 2025. — 311 с. — (Высшее образование). — URL: <https://urait.ru/bcode/561966> (дата обращения: 07.03.2025). — ISBN 978-5-534-00799-2. — Текст : электронный.
3. Веремей, Е. И. Линейные системы с обратной связью : учебное пособие / Е. И. Веремей. — Санкт-Петербург : Лань, 2022. — 448 с. — URL: <https://e.lanbook.com/book/212201> (дата обращения: 07.03.2025). — Режим доступа: для авториз. пользователей. — ISBN 978-5-8114-1412-3. — Текст : электронный.
4. Певзнер, Л. Д. Теория систем управления : учебное пособие для вузов / Л. Д. Певзнер. — 3-е изд., испр. — Санкт-Петербург : Лань, 2025. — 448 с. — URL: <https://e.lanbook.com/book/437243> (дата обращения: 07.03.2025). — Режим доступа: для авториз. пользователей. — ISBN 978-5-507-49635-8. — Текст : электронный.
5. Дорф, Р. Современные системы управления /Р. Дорф, Р. Бишоп. — М. : Лаборатория базовых знаний, 2004. — 832 с. — Текст : непосредственный.
6. Моделирование движения динамических управляемых объектов : методические указания к выполнению лабораторной работы по дисциплине "Основы управления в технических системах" для студентов направления подготовки 27.03.04 – Управление в технических системах / Севастопольский государственный университет, Институт радиоэлектроники и интеллектуальных технических систем, кафедра "Информатика и управление в технических системах" ; сост.: В. А. Карапетьян, В. А. Крамарь. – Севастополь : СевГУ, 2025. – 21 с.
7. Карапетьян, В. А. Средства Matlab в задачах анализа и синтеза линейных стационарных САУ : учебное пособие / В. А. Карапетьян, В. А. Крамарь. — 2-е изд., испр. — Москва : Центркаталог, 2021. — 240 с. — URL: <https://e.lanbook.com/book/223733> (дата обращения: 07.03.2025). — Режим доступа: для авториз. пользователей. — ISBN 978-5-903268-53-5. — Текст : электронный.
8. Symbolic Math Toolbox. User's Guide. – [N. l.] : The MathWorks, Inc., 2024. – 2400 p.
9. Matlab. Mathematics. – [N. l.] : The MathWorks, Inc., 2024. – 855 p.
10. Simulink. User's Guide. – [N. l.] : The MathWorks, Inc., 2024. – 4412 p.
11. Simulink Control Design. User's Guide. – [N. l.] : The MathWorks, Inc., 2024. – 2964 p.
12. Control System Toolbox. User's Guide. – The MathWorks, Inc., 2024. – 1964 p.

ПРИЛОЖЕНИЕ А

Пример решения основных задач лабораторной работы

Продолжим рассмотрение примера из Приложения А [6] в контексте выполнения задач текущей лабораторной работы. В качестве исследуемого ОУ выбрана модель обычной маятниковой системы на тележке с электрическим мотором. Схема этой системы показана на рисунке А.1.



Стержень маятника массой m_p и моментом инерции J_p укреплен на шарнире к движущейся вдоль оси x_0 тележке. Центр масс маятника расположен на расстоянии l_p от точки крепления и имеет координаты (x_p, y_p) в декартовой системе координат (x_0, y_0) . Угол поворота маятника α отсчитывается в положительном направлении от строго вертикального нижнего положения. Положительное направление движения тележки – слева направо.

Математическая модель движения маятниковой системы, изображенной на рисунке А.1, была получена в [6] на основе методологии Эйлера - Лагранжа.

Рисунок А.1 – Схема маятниковой системы

Эта модель в векторно-матричной форме представлена двумя нелинейными ОДУ второго порядка

$$\underbrace{\begin{bmatrix} m_c + m_p & m_p l_p \cos \alpha \\ m_p l_p \cos \alpha & J_p + m_p l_p^2 \end{bmatrix}}_{\text{Матрица } M(\alpha)} \begin{bmatrix} \ddot{x}_c \\ \ddot{\alpha} \end{bmatrix} + \underbrace{\begin{bmatrix} b_c & -m_p l_p \dot{\alpha} \sin \alpha \\ 0 & b_\alpha \end{bmatrix}}_{\text{Матрица } C(\alpha, \dot{\alpha})} \begin{bmatrix} \dot{x}_c \\ \dot{\alpha} \end{bmatrix} + \underbrace{\begin{bmatrix} 0 \\ m_p g l_p \sin \alpha \end{bmatrix}}_{\text{Матрица } G(\alpha)} = \underbrace{\begin{bmatrix} 1 \\ 0 \end{bmatrix}}_B f_c. \quad (\text{A.1})$$

Разрешив систему (А.1) относительно вектора вторых производных, перепишем (А.1) в следующем виде

$$\begin{bmatrix} \ddot{x}_c \\ \ddot{\alpha} \end{bmatrix} = -H \cdot \begin{bmatrix} b_c & -m_p l_p \dot{\alpha} \sin \alpha \\ 0 & b_\alpha \end{bmatrix} \begin{bmatrix} \dot{x}_c \\ \dot{\alpha} \end{bmatrix} - H \cdot \begin{bmatrix} 0 \\ m_p g l_p \sin \alpha \end{bmatrix} + H \cdot \begin{bmatrix} f_c \\ 0 \end{bmatrix}. \quad (\text{A.2})$$

Здесь матрица H вычисляется обращением матрицы инерций M [6].

$$H = \begin{bmatrix} m_c + m_p & m_p l_p \cos \alpha \\ m_p l_p \cos \alpha & J_p + m_p l_p^2 \end{bmatrix}^{-1} = \frac{1}{\Delta(\alpha)} \begin{bmatrix} J_p + m_p l_p^2 & -m_p l_p \cos \alpha \\ -m_p l_p \cos \alpha & m_c + m_p \end{bmatrix} = \begin{bmatrix} H_{11} & H_{12} \\ H_{21} & H_{22} \end{bmatrix}.$$

где $\Delta(\alpha) = m_c + m_p \quad J_p + m_p l_p^2 - m_p^2 l_p^2 \cos^2 \alpha$ – определитель матрицы инерций M .

После введения вектора состояния

$$x(t) = x_1(t) \mid x_2(t) \mid x_3(t) \mid x_4(t)^T = x_c(t) \mid \alpha(t) \mid \dot{x}_c(t) \mid \dot{\alpha}(t)^T; \quad u(t) = f_c(t),$$

уравнения (А.2) можно представить в виде системы ОДУ в нормальной форме:

$$\frac{d}{dt} \begin{bmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \\ x_4(t) \end{bmatrix} = \begin{bmatrix} x_3 \\ x_4 \\ \frac{m_p l_p \cos x_2 (b_a x_4 + g l_p m_p \sin x_2) - (b_c x_3 - m_p l_p x_4^2 \sin x_2)(m_p l_p^2 + J_p) + (J_p + m_p l_p^2) u}{\Delta(x_2)} \\ \frac{m_p l_p \cos x_2 (b_c x_3 - m_p l_p x_4^2 \sin x_2) - (b_a x_4 + g l_p m_p \sin x_2)(m_c + m_p) - m_p l_p \cos x_2 \cdot u}{\Delta(x_2)} \end{bmatrix}. \quad (A.3)$$

Правые части систем (A.2) и (A.3) непосредственно могут быть использованы для проведения процедуры линеаризации в соответствии с выражениями для матриц Якоби (2.6).

Пример программной реализации процесса линеаризации и построения собственного движения полученной ЛСС приведён ниже.

% Маятник на тележке	
% X=[xc alpha Dxc Dalpha]' - вектор состояния	
clc; clear;	
n=4; m=1; p=2;	% 1
% Значения параметров маятниковой системы	
mc=0.37; % - масса тележки [кг]	% 2
mp=0.127; % - масса груза [кг]	
bc=0.0005; % - коэффициент вязкого трения тележки [Н/м/сек]	
ba=0.0024; % - коэффициент вязкого трения в маятнике [Н*м/(rad/сек)]	
lp=0.1778; % - расстояние между осью маятника ЦМ [м]	
Jp=1.1987e-3; % - момент инерции массы маятника [кг*м^2]	
g = 9.81; % - ускорение свободного падения [м/сек^2]	% 3
syms xc alpha Dxc Dalpha fc	% 4
M_sym=[mc+mp mp*lp*cos(alpha) mp*lp*cos(alpha) Jp+mp*lp^2];	% 5
H_sym=M_sym^-1;	% 6
C_sym=[bc -mp*lp*sin(alpha)*Dalpha 0 ba];	% 7
G_sym=[0 mp*g*lp*sin(alpha)];	% 8
B_sym=[1 0];	% 9
RP_sym=-H_sym*(C_sym*[Dxc;Dalpha]+G_sym)+H_sym*B_sym*fc;	%10
RP_Jac=[Dxc;Dalpha;RP_sym(1);RP_sym(2)];	%11
A_sym=jacobian(RP_Jac,[xc alpha Dxc Dalpha]);	%12
B_sym=jacobian(RP_Jac,fc);	%13
xc0=0; alpha0=0; Dxc0=0; Dalpha0=0; fc0=0;	%14
A=double(subs(A_sym,{xc alpha Dxc Dalpha fc},... {xc0 alpha0 Dxc0 Dalpha0 fc0}));	%15
B=double(subs(B_sym,{xc alpha Dxc Dalpha fc},... {xc0 alpha0 Dxc0 Dalpha0 fc0}));	%16
C=[1 0 0 0; 0 1 0 0];	%17
D=zeros(2,1);	%18
sys=ss(A,B,C,D);	%19
X0=[0; deg2rad(5); 0; deg2rad(0)];	%20

<code>t_end=15;</code>	%21
<code>[y,t]=initial(sys,X0,t_end);</code>	%22
<code>subplot(1,2,1); plot(t,y(:,1)); grid on; title('Путь тележки x(t)');</code>	%23
<code>xlabel('время (сек)'); ylabel('перемещение (м)');</code>	
<code>subplot(1,2,2); plot(t,y(:,2)*180/pi); grid on;</code>	%24
<code>title('Маятник \alpha(t)');</code>	
<code>xlabel('время (сек)'); ylabel('угол (градус)');</code>	

Пояснения к тексту программы:

- 1 – установка размерностей векторов состояния, входа и выхода;
- 2 ÷ 3 – задание коэффициентов маятниковой системы;
- 4 – введение символьных переменных для x_c , α , $\dot{x}_c$, $\dot{\alpha}$ и f_c ;
- 5-9 – формирование в символьном виде матриц инерций (**M_sym**), демпфирования (затухания) (**C_sym**), гравитационных сил (**G_sym**) и внешних воздействий (**B_sym**);
- 10 – формирование вектора правых частей системы нелинейных ОДУ второго порядка (A.2);
- 11 – формирование вектора правых частей системы нелинейных ОДУ в нормальной форме (A.3);
- 12, 13 – выполнение процедуры линеаризации вектора правых частей системы ОДУ (A.3) в окрестности точки покоя с помощью функции **jacobian**. Определение символьных выражений для матриц **A** и **B** модели пространства состояний линеаризованной модели;
- 14 – задание координат точки линеаризации – состояние покоя маятниковой системы (устойчивое состояние равновесия);
- 15-18 – определение численных значений матриц ЛСС – **A**, **B**, **C** и **D**;
- 19 – формирование линеаризованной модели **sys** маятниковой системы в форме LTI-объекта Matlab;
- 20 – задание начального возмущённого состояния линеаризованной системы **X0** в малой окрестности точки линеаризации (величина отклонения маятника $+5^\circ$);
- 21 – задание длительности интервала моделирования ЛСС;
- 22 – моделирование собственного движения ЛСС на интервале $[0; t_{\text{end}}]$ с начальными условиями **X0** и сохранение результатов расчётов в массивах **t** и **y**;
- 23, 24 – построение графиков перемещения тележки $x_c(t)$ и угла маятника $\alpha(t)$.

Результаты работы программы изображены на графиках собственного движения линеаризованной системы маятниковой системы:

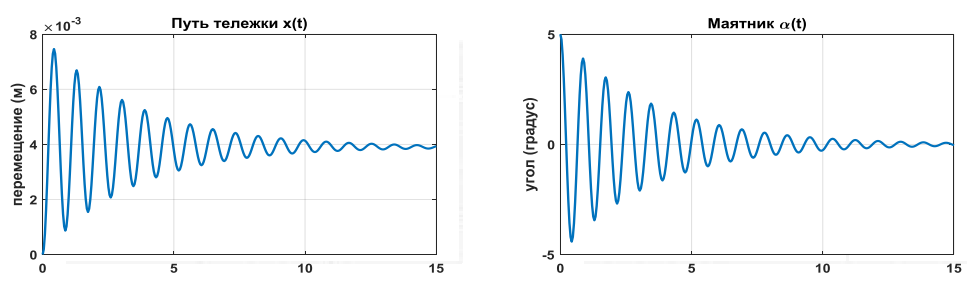


Рисунок А.2 – Графики собственного движения линеаризованной системы

Полученные в программе матрицы линеаризованной, в окрестности точки покоя, модели маятниковой представлены ниже:

A =	B =	C =	D =
0 0 1.0 0	0	1 0 0 0	0
0 0 0 1.0	0	0 1 0 0	0
0 2.4034 -0.0013 0.026	2.505		
0 -52.897 0.0054 -0.5731	-10.8496		

Для проведения процедуры линеаризации в пакете Simulink построена Simulink-схема (рисунок А.3) нелинейной математической модели движения маятниковой системы по уравнениям (А.1). Предварительно векторно-матричная модель (А.1) была преобразована в скалярную форму:

$$\begin{cases} (m_c + m_p) \ddot{x}(t) + m_p l_p \cos a \cdot \ddot{a}(t) + b_c \dot{x}(t) - m_p l_p \dot{a} \sin a \cdot \dot{a}(t) = f_c, \\ m_p l_p \cos a \cdot \ddot{x}(t) + (J_p + m_p l_p^2) \ddot{a}(t) + b_a \dot{a}(t) + m_p g l_p \sin a = 0. \end{cases}$$

Из первого уравнения полученной системы уравнений явно выражена вторая производная линейного перемещения тележки $\ddot{x}(t)$, а из второго уравнения – угловое ускорение маятника $\ddot{a}(t)$:

$$\begin{cases} \ddot{x}(t) = \frac{1}{m_c + m_p} [-m_p l_p \cos a \cdot \ddot{a}(t) - b_c \dot{x}(t) + m_p l_p \dot{a} \sin a \cdot \dot{a}(t) + f_c], \\ \ddot{a}(t) = \frac{1}{J_p + m_p l_p^2} [-m_p l_p \cos a \cdot \ddot{x}(t) - b_a \dot{a}(t) - m_p g l_p \sin a]. \end{cases} \quad (\text{A.4})$$

Последняя система уравнений реализована в виде Simulink-схемы, представленной на рисунке А.3.

Схема использована как при получении параметров линеаризованной системы в окрестности заданной рабочей точки (функция **linmod** и инструмент Model Linearizer), так и для моделирования собственного движения нелинейной системы и построения графиков выходных переменных $x_c(t)$ и $a(t)$.

На схеме блоки, выделенные оранжевым цветом, формируют первое уравнение системы (А.4) – дифференциальное уравнение перемещения тележки, а блоки зелёного цвета – второе уравнение из (А.4) – дифференциальное уравнение движения маятника.

Всем основным блокам схемы присвоены соответствующие идентификаторы – четыре интегратора $\{x(t), Dx(t) \text{ и } \alpha(t), D\alpha(t)\}$, выходы которых представляют собой переменные состояния $x_c(t) - x$, $a(t) - \alpha$, $\dot{x}(t) - Dx$ и $\dot{a}(t) - D\alpha$. Коэффициенты блоков усилителей (**Gain**) приведены в виде символьных выражений $\{M=mc, m=mp, L=lp, J=Jp, b=bc, c=ba\}$ в терминах коэффициентов уравнений (А.4). Два сумматора (малиновый **Sum_1** и голубой **Sum_2**) указывают на явную взаимосвязь уравнений модели через вторые производные переменных $x(t)$ и $a(t)$.

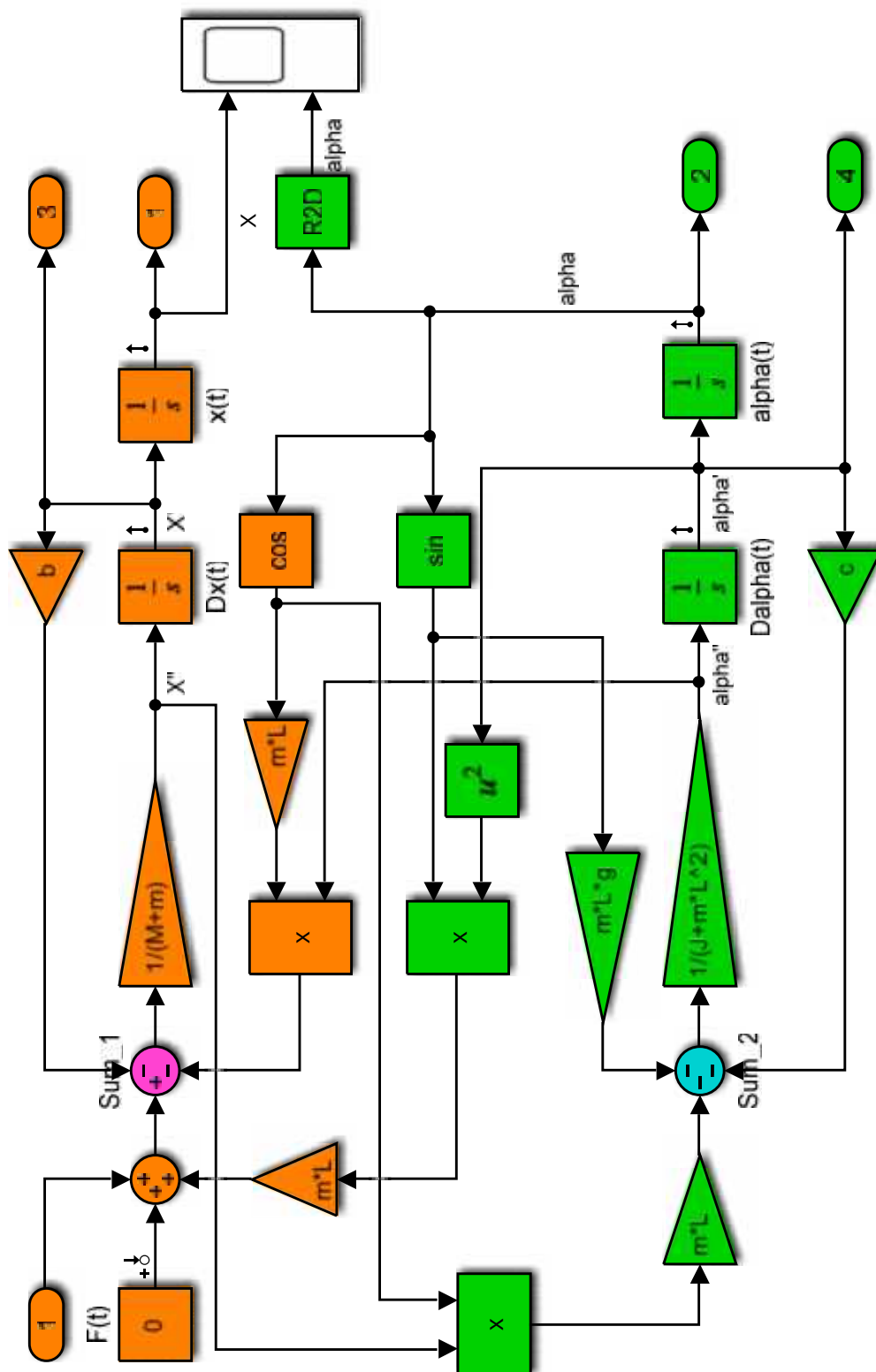


Рисунок А.3 – Simulink-схема нелинейной математической модели движения маятниковой системы

В схеме обозначены один блок-вход (блок **Inport**) и четыре блока-выхода (блоки **Outport**). Вход схемы (выходной сигнал блока-Constant **F(t)**) обозначен как сигнал-вход типа Input Perturbation , а выходные сигналы всех интеграторов – как сигналы-измерения типа Output Measurement .

Наличие блока **Scope** в схеме предусмотрено для визуализации процессов на выходе $[x_c(t) \ a(t)]$ нелинейной системы при моделировании её собственного движения. На рисунке A.4 приведены графики выхода нелинейной системы при моделировании её собственного движения для возмущённых начальных условий, аналогичных начальным условиям, назначенным в приведённой выше программе моделирования $x_0=[0; \text{deg2rad}(5); 0; \text{deg2rad}(0)]$. Графики, построенные в Matlab-программе (рисунок A.2) и графики, построенные Simulink-моделью в **Scope** (рисунок A.4), полностью совпадают. Данный факт с "большой вероятностью" (highly likely) подтверждает эквивалентность двух способов решения уравнений нелинейной модели маятниковой системы (A.4).

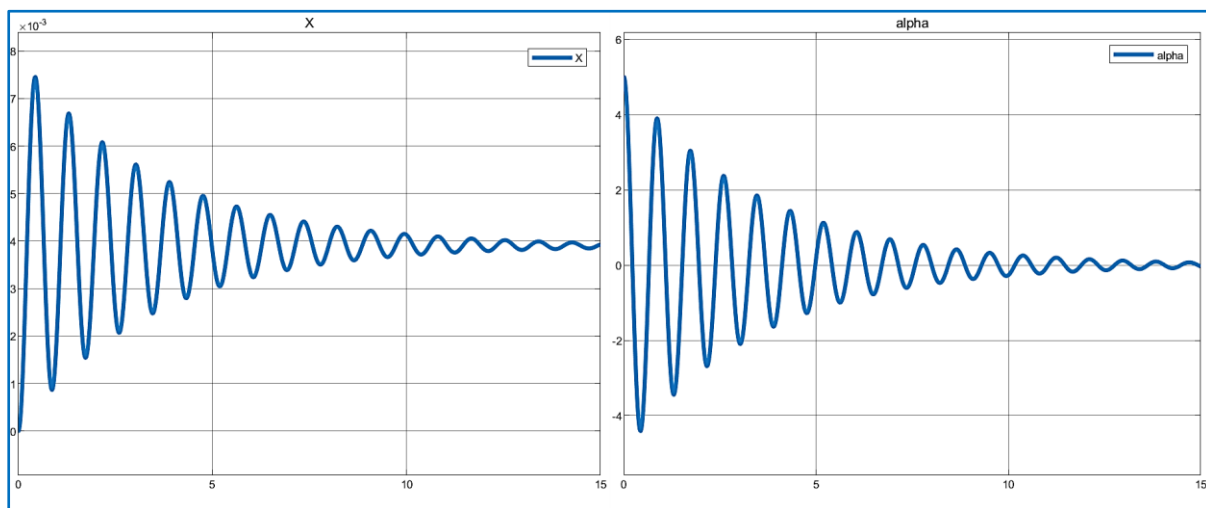


Рисунок A.4 – Simulink-графики собственного движения нелинейной системы

Параметры линеаризованной математической модели движения системы в малой окрестности рабочей точки получены с помощью функции **linmod**, вызванной следующей инструкцией:

```
[A,B,C,D]=linmod('Pendulum_Cart_Simul',[0 0 0 0],[0])
```

Входные фактические параметры функции означают:

- **'Pendulum_Cart_Simul'** – имя Simulink-файла с Simulink-схемой динамической модели, предназначенной для линеаризации (схема рисунка A.3);
- **[0 0 0 0]** – координаты рабочей точки в пространстве состояния маятниковой системы, в окрестности которой проводится линеаризация. В нашем случае – это точка покоя маятниковой системы – неподвижное $\{ x_c(t) \int 0; a(t) \int 0 \}$ состояние тележки и маятника, расположенных в начале координат по положению $x_c(t) \int 0$ и углу отклонения $a(t) \int 0$;
- **[0]** – значение входного воздействия $f_c(t)=u(t) \int 0$.

Выходными параметрами функции являются матрицы коэффициентов модели пространства состояния (2.7): **A**, **B**, **C** и **D**.

После отработки, приведённой выше инструкции, в командное окно Matlab выводятся значения матриц ЛСС маятниковой системы:

$$\begin{aligned}
 \mathbf{A} &= \begin{bmatrix} 0 & 0 & 1.0000 & 0 \\ 0 & 0 & 0 & 1.0000 \\ 0 & 2.4034 & -0.0013 & 0.0260 \\ 0 & -52.8979 & 0.0054 & -0.5731 \end{bmatrix} & \mathbf{B} &= \begin{bmatrix} 0 \\ 0 \\ 2.5050 \\ -10.8496 \end{bmatrix} \\
 \mathbf{C} &= \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} & \mathbf{D} &= \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}
 \end{aligned} \tag{A.5}$$

Для дальнейшей корректной работы с ЛСС (A.5), следует привести матрицы уравнения выхода модели к виду, соответствующему физическому содержанию исследуемой системы, а именно: управляемым вектором выхода является вектор $y(t)=[x_c(t) \ a(t)]^T$ и поэтому матрицы **C** и **D** должны иметь следующий вид:

$$\begin{aligned}
 \mathbf{C} &= \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} & \mathbf{D} &= \begin{bmatrix} 0 \\ 0 \end{bmatrix}
 \end{aligned}$$

Для получения линеаризованной модели в форме передаточной функции (матричной передаточной функции), инструкция с функцией **linmod** должна быть сформирована, например, так:

[NUM,DEN]=linmod('Pendulum_Cart_Simul',[0 0 0 0],[0])

Выходными параметрами здесь будут:

- **NUM** – числовая матрица размерности $p \times (n+1)$, у которой каждая строка представляет коэффициенты полиномов *числителей* соответствующих частных ПФ;
- **DEN** – числовая матрица размерности $p \times (n+1)$, у которой каждая строка представляет коэффициенты полиномов *знаменателей* соответствующих частных ПФ. Если полиномы всех знаменателей одинаковы, то **DEN** возвращается в виде одной строки коэффициентов.

Обработка рассматриваемой инструкции вернёт в командное окно Matlab следующую информацию о ЛСС в форме ПФ:

$$\begin{aligned}
 \mathbf{NUM} &= \begin{bmatrix} 0 & 0 & 2.5050 & 1.1532 & 106.4344 \\ 0 & 0 & -10.8496 & 0.0000 & 0.0000 \\ 0 & 2.5050 & 1.1532 & 106.4344 & 0 \\ 0 & -10.8496 & 0 & 0 & 0 \end{bmatrix} \\
 \mathbf{DEN} &= \begin{bmatrix} 1.0000 & 0.5744 & 52.8985 & 0.0532 & 0 \end{bmatrix}
 \end{aligned}$$

Структура массивов **NUM** и **DEN** определяет матричную ПФ (вектор-столбец 4×1), состоящую из четырёх ПФ, связывающих единственный управляющий вход системы $u=fc$ и четыре компоненты вектора состояния $[x_c \ a \ x_\varphi \ a]^T$:

$$W_{xu}(s) = \begin{bmatrix} W_{x_u}(s) \\ W_{a_u}(s) \\ W_{\dot{x}_u}(s) \\ W_{\dot{a}_u}(s) \end{bmatrix} = \begin{bmatrix} W_{11}(s) \\ W_{21}(s) \\ W_{31}(s) \\ W_{41}(s) \end{bmatrix} = \frac{1}{s^4 + 0.5744s^3 + 52.8985s^2 + 0.0532s} \begin{bmatrix} 2.505s^2 + 1.1532s + 106.4344 \\ -10.8496s^2 \\ 2.505s^3 + 1.1532s^2 + 106.4344s \\ -10.8496s^3 \end{bmatrix}.$$

Значимые ПФ в нашем случае: $W_{x_u}(s)$ и $W_{a_u}(s)$:

$$W_{yu}(s) = \begin{bmatrix} W_{x_u}(s) \\ W_{a_u}(s) \end{bmatrix} = \begin{bmatrix} W_{11}(s) \\ W_{21}(s) \end{bmatrix} = \begin{bmatrix} \frac{2.505s^2 + 1.1532s + 106.4344}{s^4 + 0.5744s^3 + 52.8985s^2 + 0.0532s} \\ \frac{-10.8496s^2}{s^4 + 0.5744s^3 + 52.8985s^2 + 0.0532s} \end{bmatrix}.$$

Сформируем LTI-объект линеаризованной системы в виде модели в пространстве состояний. Для этого используем функцию **ss** из пакета Control System Toolbox:

sys_ss=ss(A,B,C,D)

Исполнение данной команды приведёт к созданию LTI-объекта **sys_ss** и выводу в командное окно Matlab следующей информации:

sys_ss =

A =

	x1	x2	x3	x4
x1	0	0	1	0
x2	0	0	0	1
x3	0	2.403	-0.001253	0.02604
x4	0	-52.9	0.005425	-0.5731

B =

	u1
x1	0
x2	0
x3	2.505
x4	-10.85

C =

	x1	x2	x3	x4
y1	1	0	0	0
y2	0	1	0	0

D =

	u1
y1	0
y2	0

Continuous-time state-space model.

Расчёт и построение графиков собственного движения ЛСС **sys_ss** для начальных условий $X_0 = [0; \text{deg2rad}(5); 0; \text{deg2rad}(0)]$ представлено последовательностью инструкций 20–24 из приведённой выше Matlab программы.

Формирование LTI-объекта в виде матричной передаточной функции осуществим с помощью функции **tf** из пакета Control System Toolbox:

sys_tf= tf({NUM(1,:);NUM(2,:)},DEN)

Будет создан SIMO LTI-объект **sys_tf**, а в командное окно Matlab – выведена следующая информация об этой ЛСС:

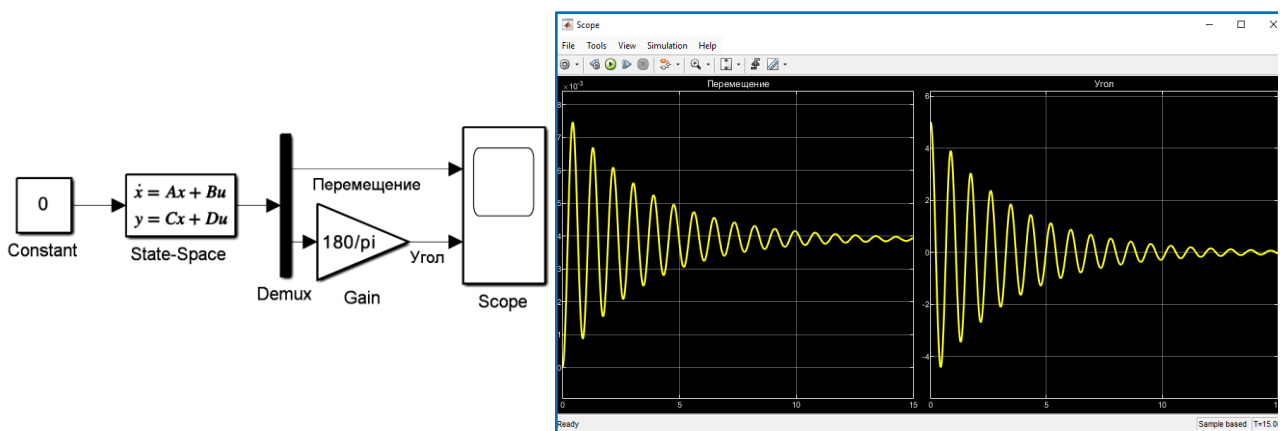
sys_tf =

From input to output...

$$\begin{array}{l} 2.505 s^2 + 1.153 s + 106.4 \\ 1: \text{-----} \\ s^4 + 0.5744 s^3 + 52.9 s^2 + 0.05322 s \\ \\ -10.85 s^2 \\ 2: \text{-----} \\ s^4 + 0.5744 s^3 + 52.9 s^2 + 0.05322 s \end{array}$$

Continuous-time transfer function.

Построение графиков собственного движения в Simulink осуществим с помощью, например, такой Simulink-схемы:



Для применения средств Model Linearizer сигнал из блока-источника $F(t)$ (рис. A3) следует пометить как Input Perturbation (выделить правой кнопкой манипулятора исходящую из $F(t)$ сигнальную линию и в контекстно-зависимом меню выбрать опции Linear Analysis Points > Input Perturbation), а выходные сигналы Simulink-схемы (входы четырёх блоков Outport) следует пометить через меню Linear Analysis Points как Output Measurement. Так как линеаризация проводится в окрестности точки покоя маятниковой системы (в нижнем вертикальном положении), то начальные условия для всех четырёх блоков-интеграторов схемы должны быть нулевыми. В таком состоянии применение инструмента Model Linearizer для получения параметров линеаризованной системы сводится к нескольким простейшим манипуляциям:

- выбор в главном меню окна Simulink последовательности опций
APPS > Model Linearizer...
- в появившемся окне средства Model Linearizer в закладке LINEAR ANALYSIS в разделе OPTIONS следует выбрать флажок Result Viewer;
- нажать, например, кнопку Impulse в разделе LINEARIZE ленты.

Ниже на рисунке A.5 представлен результат наших манипуляций. Здесь изображены два окна: основное – Model Linearizer и вспомогательное (модальное) – Linearization result details for linsys1. В основном окне слева в подокне

Linear Analysis Workspace появился объект **linsys1**. Это идентификатор полученной линеаризованной LTI-системы. Выбрав объект **linsys1** правой кнопкой мышки, с помощью опции **Export to MATLAB Workspace** можно передать этот LTI-объект в рабочую область Matlab. После этого LTI-объект станет доступным для дальнейшей работы с ним в Matlab.

Вспомогательное окно **Linearization result details for linsys1** содержит данные о значениях всех матриц линеаризованной системы, представленной в форме пространства состояний. В верхней части это окна расположен элемент управления типа "выпадающий список" **Display Linearization result as:** с помощью которого можно вывести во вспомогательное окно параметры линеаризованной системы в формах **zpk**- или **tf**-моделей.

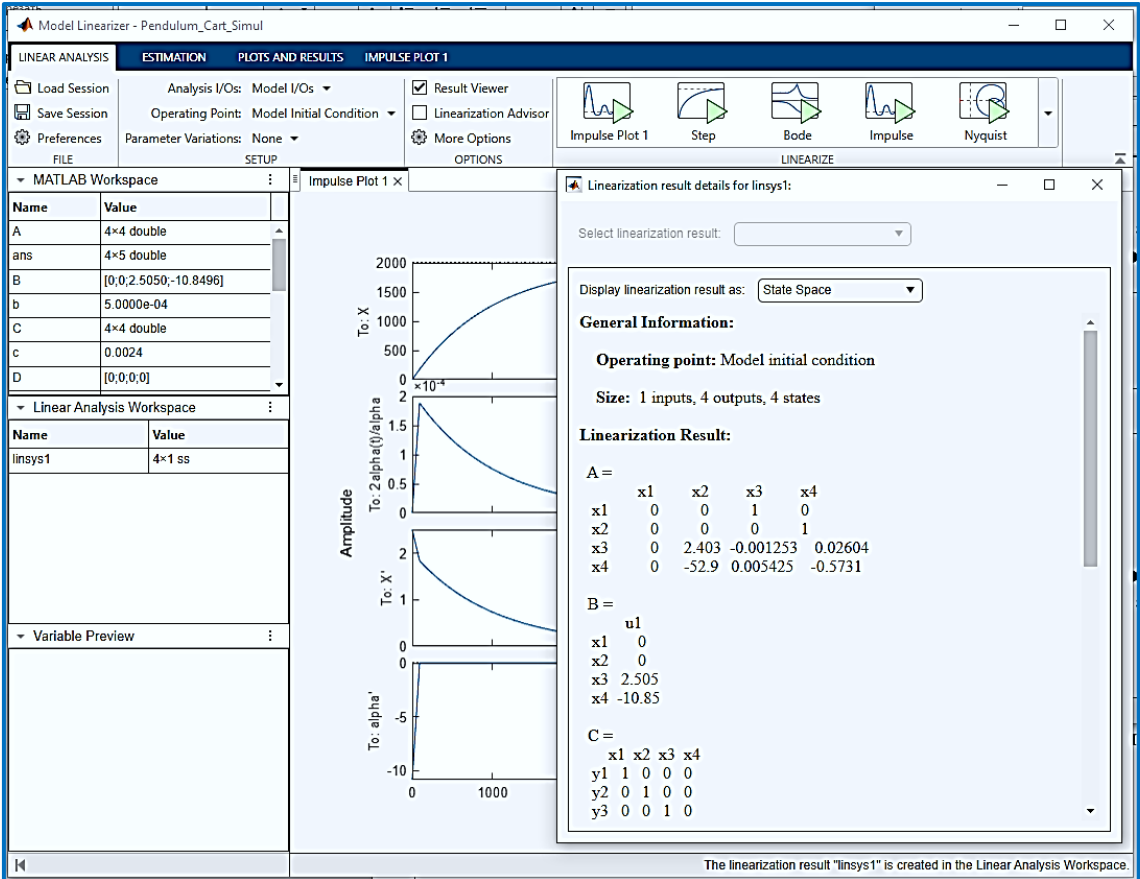


Рисунок А.5 – Результаты линеаризации в среде Model Linearizer

ЛИНЕАРИЗАЦИЯ МОДЕЛЕЙ ДИНАМИЧЕСКИХ УПРАВЛЯЕМЫХ ОБЪЕКТОВ

Методические указания

Составители: **Карапетьян** Валерий Артёмович,
Крамарь Вадим Александрович

В авторской редакции

Изд. № 46/2025. Объем 1,5 п.л. Усл. печ. л. 1,39. Уч.-изд. л. 1,47.
Издательский центр ФГАОУ ВО «Севастопольский государственный университет»