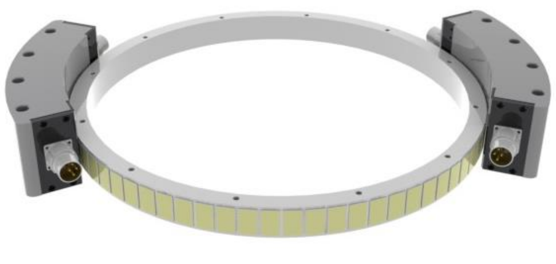
Министерство образования и науки российской федерации
Федеральное государственное бюджетное образовательное учреждение
высшего профессионального образования
«ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ СИСТЕМ УПРАВЛЕНИЯ И РАДИОЭЛЕКТРОНИКИ» (ТУСУР)

Утверждаю
Зав. отделением каф. ЮНЕСКО
_________________ Ю.М. Осипов

"_____"_____________________ 2013 г.

СТАТИСТИЧЕСКАЯ ДИНАМИКА СИСТЕМ АВТОМАТИЧЕСКОГО УПРАВЛЕНИЯ

Методические указания к лабораторным работам
по дисциплине «**Исследование динамики электромехатронных систем движения**» для магистрантов 1 курса, обучающихся по направлению 221000.68 «Мехатроника и робототехника» по магистерским программам «Компьютерное моделирование электромехатронных систем движения" и «Проектирование и исследование мультикоординатных электромехатронных систем движения»



Томск 2013

УДК 621.396.6.671.7

Статистическая динамика систем автоматического управления: Методические указания к лабораторным работам по дисциплине «Исследование динамики электромехатронных систем движения» для магистрантов 1 курса, обучающихся по направлению 221000.68 «Мехатроника и робототехника» по магистерским программам «Компьютерное моделирование электромехатронных систем движения" и «Проектирование и исследование мультикоординатных электромехатронных систем движения». - Томск: Изд-во ТУСУР, 2013. – 52 с.

Методические указания рассмотрены и рекомендованы к изданию методическим семинаром отделения кафедры ЮНЕСКО
« 26 » августа 2013 г.

Составители:
к.э.н., доц. _____________ О.Ю. Осипов

к.т.н., доц. ______________С.В. Щербинин

Зав. кафедрой ОКЮ
доктор техн. наук,
доктор экон. наук
профессор _______________ *Ю.М. Осипов*

ВВЕДЕНИЕ

Данное пособие предназначено для самостоятельного проведения студентами цикла лабораторных работ по курсу «Статистическая динамика систем автоматического управления» на ПЭВМ с использованием программного пакета MATLAB. Такой курс предусмотрен в учебном плане ряда специальностей, связанных с автоматизацией, мехатроникой и робототехникой. Предполагается, что студенты знакомы с основным курсом теории автоматического управления и с теорией вероятностей. Для того чтобы облегчить проведение лабораторных работ, в методических указаниях даны основные соотношения из курса «Статистическая динамика систем автоматического управления», которые используются при статистическом анализе линейных стационарных автоматических систем. Приведено краткое описание возможностей пакета MATLAB 7 для проведения такого анализа.

В пособии даны указания к проведению трех лабораторных работ, объединенных в общий цикл. Вначале нужно построить формирующий фильтр, позволяющий получить случайный процесс с заданными статистическими характеристиками. Затем провести исследование системы, на вход которой поступают детерминированный сигнал и случайная помеха. Наконец, предлагается исследовать замкнутую систему автоматического регулирования, ко входу которой приложен случайный сигнал, представляющий собой аддитивную смесь полезного случайного сигнала и случайного шума. Описание каждой лабораторной работы сопровождается примерами. Для того чтобы правильно интерпретировать получаемые результаты, мы рекомендуем предварительно выполнить статистическое исследование в аналитической форме с использованием приведенных в пособии соотношений. Более подробную информацию о методах статистического анализа автоматических систем и о возможностях пакета MATLAB можно найти в литературе, список которой приведен в конце пособия [1 – 4].

1. ОСНОВНЫЕ МАТЕМАТИЧЕСКИЕ СООТНОШЕНИЯ, ИСПОЛЬЗУЕМЫЕ ПРИ СТАТИСТИЧЕСКОМ АНАЛИЗЕ АВТОМАТИЧЕСКИХ СИСТЕМ

1.1. Характеристики случайных процессов

Основными характеристиками случайных величин являются функция распределения вероятности и плотность распределения вероятности. Функция распределения случайной величины ξ, принимающей любые вещественные значения, определяется соотношением

$$F_\xi(x) = P\{\xi < x\} \tag{1}$$

и представляет собой вероятность того, что случайная величина ξ принимает значения, меньшие заданного значения x.

Плотность распределения вероятности случайной величины ξ может быть определена по функции распределения вероятности с использованием формулы

$$p_\xi(x) = \frac{dF_\xi(x)}{dx}. \tag{2}$$

Случайный процесс определяется множеством случайных реализаций $\xi(t)$, $0 \le t \le T$. Фиксируя произвольным образом моменты времени t_i, $i = 1, 2, \ldots, N$, можно получить N-мерную случайную величину $\boldsymbol{\xi} = [\xi(t_1), \xi(t_2), \ldots, \xi(t_N)]$, т. е. случайный вектор, компонентами которого являются случайные величины, представляющие собой значения реализаций $\xi(t)$ в дискретные моменты времени. Таким образом, случайный процесс характеризуется множеством функций распределения вероятности, определяющих векторную случайную величину $\boldsymbol{\xi}$:

$$F_{\xi k}(x_1, x_2, ..., x_k, t_1, t_2, ..., t_k) = P\{\xi(t_1) < x_1, \quad \xi(t_2) < x_2, ..., \xi(t_k) < x_k\}, \quad (3)$$

где k = 1, 2, …, N, или соответствующим множеством плотностей распределения вероятности:

$$p_{\xi k}(x_1, x_2, \ldots, x_k, t_1, t_2, \ldots, t_k) = \\ = \frac{\partial^k}{\partial x_1 \partial x_2 \ldots \partial x_k} F_{\xi k}(x_1, x_2, \ldots, x_k, t_1, t_2, \ldots, t_k), \quad (4)$$

где k = 1, 2, …, N.

Используя плотности распределения вероятности, можно определить моменты различного порядка для случайного процесса $\xi(t)$. Наиболее часто применяют начальный момент первого порядка (математическое ожидание):

$$m_\xi(t) = M\{\xi(t)\} = \int_{-\infty}^{\infty} x p_{\xi 1}(x, t) dx \quad (5)$$

и центральный момент второго порядка (корреляционную (автокорреляционную) функцию):

$$K_\xi(t_1, t_2) = M\{\overset{\circ}{\xi}(t_1)\overset{\circ}{\xi}(t_2)\} = \\ = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty}(x_1 - m_\xi(t_1))(x_2 - m_\xi(t_2)) p_{\xi 2}(x_1, x_2, t_1, t_2) dx_1 dx_2. \quad (6)$$

Из последнего выражения можно также найти дисперсию случайного процесса:

$$D_\xi(t) = M\{\overset{\circ}{\xi}(t)\overset{\circ}{\xi}(t)\}. \quad (7)$$

Взаимнокорреляционную функцию двух случайных процессов $\xi(t)$ и $\eta(t)$ определяют по формуле

$$K_{\xi\eta}(t_1, t_2) = M\{\overset{\circ}{\xi}(t_1)\overset{\circ}{\eta}(t_2)\} = \\ = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty}(x_1 - m_\xi(t_1))(x_2 - m_\xi(t_2)) p_{\xi\eta 2}(x_1, x_2, t_1, t_2) dx_1 dx_2. \quad (8)$$

Напомним, что случайный процесс $\xi(t)$ называется стационарным (в широком смысле), если его математическое ожидание постоянно, а корреляционная функция зависит только от разности аргументов:

$$m_{\xi}(t) = m_{\xi} = \mathrm{const}, \tag{9}$$

$$K_{\xi}(t_1, t_2) = K_{\xi}(t_1 - t_2) = K_{\xi}(\tau). \tag{10}$$

Для стационарных процессов можно определить спектральную плотность случайного процесса как преобразование Фурье корреляционной функции:

$$S_{\xi}(\omega) = \int_{-\infty}^{\infty} K_{\xi}(\tau) e^{-j\omega\tau} d\tau. \tag{11}$$

По заданной спектральной плотности можно определить корреляционную функцию:

$$K_{\xi}(\tau) = \frac{1}{2\pi} \int_{-\infty}^{\infty} S_{\xi}(\omega) e^{j\omega\tau} d\omega. \tag{12}$$

Из последней формулы следует и выражение для вычисления дисперсии стационарного случайного процесса по его спектральной плотности:

$$D_{\xi} = K_{\xi}(0) = \frac{1}{2\pi} \int_{-\infty}^{\infty} S_{\xi}(\omega) d\omega. \tag{13}$$

При статистической обработке случайных сигналов используются методы и формулы математической статистики.

Пусть на интервале времени $[0, T]$ экспериментально получено n реализаций случайного процесса $\xi(t)$, которые мы обозначим $x_i(t)$, $i = 1, 2, \ldots, n$. Тогда оценка математического ожидания случайного процесса может быть определена по формуле

$$\hat{m}_{\xi}(t) = \frac{1}{n} \sum_{i=1}^{n} x_i(t). \tag{14}$$

Оценка (14) является несмещенной, т. е. $M\{\hat{m}_{\xi}\} = m_{\xi}$, и состоятельной, поскольку выполняется условие $\lim\limits_{n \to \infty} D\{\hat{m}_{\xi}\} = 0$.

Вычислив оценку математического ожидания случайного процесса, можно найти оценку его автокорреляционной функции по формуле

$$\hat{K}_{\xi}(t_1, t_2) = \frac{1}{n-1}\sum_{i=1}^{n}(x_i(t_1) - \hat{m}_{\xi}(t_1))(x_i(t_2) - \hat{m}_{\xi}(t_2)). \tag{15}$$

Здесь используется множитель $\frac{1}{n-1}$, а не $\frac{1}{n}$, чтобы обеспечить несмещенность оценки (15). Ее состоятельность, как и состоятельность оценки математического ожидания, можно легко проверить.

Для оценки дисперсии из последней формулы получим:

$$\hat{D}_{\xi}(t) = \frac{1}{n-1}\sum_{i=1}^{n}(x_i(t) - \hat{m}_{\xi}(t))^2. \tag{16}$$

При исследовании статистических характеристик стационарных случайных процессов часто применяют эргодическую гипотезу, позволяющую существенно упростить их вычисление. В этом случае вместо усреднения значений множества реализаций в одни и те же моменты времени, как в формулах (14)–(16), усредняют значения одной реализации, взятые в различные моменты времени.

Стационарный процесс называется эргодическим по отношению к математическому ожиданию, если является несмещенной и состоятельной следующая статистическая оценка математического ожидания, определяемая по одной его реализации $x(t)$, измеряемой на интервале времени $[0; T]$:

$$\hat{m}_{\xi} = \frac{1}{T}\int_{0}^{T} x(t)dt. \tag{17}$$

Эта оценка является несмещенной и состоятельной, если выполняются условия $M\{\hat{m}_{\xi}\} = m_{\xi}$ и $\lim\limits_{n\to\infty} D\{\hat{m}_{\xi}\} = 0.$ Нетрудно непосредственно убедиться в несмещенности оценки (17). Для ее состоятельности нужно дополнительно потребовать, чтобы выполнялось условие

$$\lim_{\tau\to\infty} K_{\xi}(\tau) = 0. \tag{18}$$

Смысл последнего условия состоит в том, что значения случайных величин $\xi(t_i)$ и $\xi(t_j)$ становятся слабо коррелированными при увеличении временного интервала $(t_j - t_i)$.

Для практических расчетов по формуле (17) используют приближенное соотношение

$$\hat{m}_\xi = \frac{1}{N+1}\sum_{i=0}^{N} x(t_i). \tag{19}$$

В частности, если интервал $[0; T]$ разбит на N элементарных интервалов длиной $\Delta t = T/N$, то $t_i = i\Delta t$, $i = 0, 1, \ldots, N$. Можно также записать, что $N = T/\Delta t = Tf_d$, где $f_d = 1/\Delta t$ — частота дискретизации, Гц.

Рассмотрим теперь оценку корреляционной функции стационарного случайного процесса:

$$K_\xi(\tau) = M\{(\xi(t) - \hat{m}_\xi)(\xi(t+\tau) - \hat{m}_\xi)\}. \tag{20}$$

Обозначив $x(t)$ отдельную реализацию случайного процесса и принимая во внимание, что интервал, на котором происходит вычисление оценки корреляционной функции, равен $[0; T - \tau]$, получим следующую формулу для оценки корреляционной функции:

$$\hat{K}_\xi(\tau) = \frac{1}{T-\tau}\int_0^{T-\tau} (x(t) - \hat{m}_\xi)(x(t+\tau) - \hat{m}_\xi)dt, \tag{21}$$

или

$$\hat{K}_\xi(\tau) = \frac{1}{T-\tau}\int_0^{T-\tau} x(t)x(t+\tau)dt - \hat{m}_\xi^2. \tag{22}$$

Эти оценки являются несмещенными, а для их состоятельности достаточно выполнения условия (18). В этом случае стационарный случайный процесс $\xi(t)$ называется эргодическим по отношению к корреляционной функции. Вычисление корреляционной функции как среднего по множеству в формуле (20) можно приближенно заменить вычислением среднего по времени согласно формуле (21) или (22). В частности, при $\tau = 0$ отсюда можно получить оценку для дисперсии эргодического процесса:

$$\hat{D}_\xi = \frac{1}{T}\int_0^T (x(t) - \hat{m}_\xi)^2 dt = \frac{1}{T}\int_0^T x^2(t)dt - \hat{m}_\xi^2. \tag{23}$$

На практике оценку корреляционной функции обычно вычисляют по дискретным значениям реализации случайного процесса $\xi(t_i)$. В этом случае вместо (21) используют следующую формулу:

$$\hat{K}_\xi(m) = \frac{1}{N-m+1}\sum_{i=0}^{N-m}(x(t_i)-\hat{m}_\xi)(x(t_{i+m})-\hat{m}_\xi), \tag{24}$$

где $m = 0, 1, 2, \ldots, N$, $N = T/\Delta t = Tf_d$.

Из последней формулы при $m = 0$ получим оценку дисперсии и стандартного (среднего квадратического) отклонения:

$$\hat{D}_\xi = \frac{1}{N+1}\sum_{i=0}^{N}(x(t_i)-\hat{m}_\xi)^2, \quad \hat{\sigma}_\varepsilon = \sqrt{\hat{D}_\varepsilon}. \tag{25}$$

Определив оценку корреляционной функции, можно вычислить и оценку спектральной плотности (11). С учетом четности корреляционной функции формулу (11) можно переписать в виде

$$S_\xi(\omega) = 2\int_0^\infty K_\xi(\tau)\cos\omega\tau\, d\tau.$$

Учитывая, что оценка корреляционной функции (24) получена на интервале $[0; T]$, получим следующую формулу для вычисления оценки спектральной плотности эргодического случайного процесса:

$$\hat{S}_\xi(\omega) = \Delta\tau\,\hat{K}_\xi(0) + 2\Delta\tau\sum_{m=1}^{N}\hat{K}_\xi(m\Delta\tau)\cos\omega m\Delta\tau, \tag{26}$$

где $\Delta\tau = T/N$.

В данном цикле лабораторных работ предполагается, что рассматриваемые случайные процессы являются стационарными и эргодическими. Поэтому их статистические характеристики могут быть определены как по формулам (14) – (16)), т. е. путем усреднения по множеству реализаций, так и по формулам (19), (24), (25), предполагающим усреднение по времени, вычисляемое для одной реализации при достаточно большом времени наблюдения.

С учетом условия (18) время наблюдения T выбирают из условия $|\hat{K}_\xi(\tau)| < \varepsilon$ при $\tau > T$, где ε — достаточно малое положительное число по сравнению с $\hat{K}_\xi(\tau)$. Поскольку максимальное значение $\hat{K}_\xi(\tau)$ обычно достигается при $\tau = 0$, то можно выбрать, например, $\varepsilon < 0{,}05\,\hat{K}_\xi(0) = 0{,}05\,\hat{D}_\xi$.

1.2. Определение характеристик стационарного случайного процесса на выходе линейной системы

Рассмотрим линейную систему с постоянными параметрами, на вход которой поступает стационарный случайный процесс $\xi(t)$. Система описывается своей передаточной функцией $W(s)$ (рис. 1).

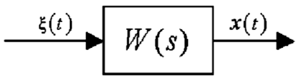


Рис. 1. Линейная система, на вход которой поступает случайный процесс

Ниже приведены формулы, позволяющие определить основные характеристики случайного процесса на выходе системы $x(t)$, если известна ее передаточная функция и соответствующие характеристики входного случайного процесса $\xi(t)$, в том числе математическое ожидание, спектральную плотность, автокорреляционную функцию и дисперсию случайного процесса $x(t)$:

$$m_x = W(0)m_\xi; \tag{27}$$

$$S_x(\omega) = \left|W(j\omega)\right|^2 S_\xi(\omega); \tag{28}$$

$$K_x(t) = \frac{1}{2\pi}\int_{-\infty}^{\infty} \left|W(j\omega)\right|^2 S_\xi(\omega) e^{j\omega t} d\omega; \tag{29}$$

$$D_x = \frac{1}{2\pi}\int_{-\infty}^{\infty} \left|W(j\omega)\right|^2 S_\xi(\omega) d\omega. \tag{30}$$

Приведенные формулы можно использовать и для формирования случайного процесса с заданными характеристиками из белого шума с помощью формирующего фильтра. Если случайный процесс $\xi(t)$ на входе фильтра с передаточной функцией $W_ф(s)$ — белый шум, для которого $m_\xi = 0$, $S_\xi(\omega) = c^2$, то в соответствии с приведенными формулами характеристики случайного процесса $x(t)$ на выходе формирующего фильтра составят соответственно

$$m_x = 0;$$

$$S_x(\omega) = \left|W_\text{ф}(j\omega)\right|^2 c^2; \qquad (31)$$

$$D_x = \frac{1}{2\pi}\int_{-\infty}^{\infty}\left|W_\text{ф}(j\omega)\right|^2 c^2 d\omega. \qquad (32)$$

Если задана спектральная плотность случайного процесса $S_x(\omega)$, то, используя формулу (31), можно подобрать частотную характеристику формирующего фильтра.

Пусть теперь на вход системы подается аддитивная смесь полезного случайного сигнала $\xi(t)$ и случайного сигнала $\eta(t)$, являющегося помехой (рис. 2), тогда дисперсия ошибки $\varepsilon(t) = \xi(t) - x(t)$ определяется следующим образом:

$$D_\varepsilon = \frac{1}{2\pi}\int_{-\infty}^{\infty} S_\xi(\omega)\left|\Phi_\varepsilon(j\omega)\right|^2 d\omega + \frac{1}{2\pi}\int_{-\infty}^{\infty} S_\eta(\omega)\left|W(j\omega)\right|^2 d\omega, \qquad (33)$$

где

$$\Phi_\varepsilon(j\omega) = \frac{1}{1+W(j\omega)}. \qquad (34)$$

Для замкнутой системы автоматического регулирования, на вход которой подается аддитивная смесь полезного случайного сигнала $\xi(t)$ и помехи $\eta(t)$ (рис. 3), дисперсия ошибки является показателем точности отработки случайного полезного сигнала и вычисляется по формуле

$$D_\varepsilon = \frac{1}{2\pi}\int_{-\infty}^{\infty} S_\xi(\omega)\left|\Phi_\varepsilon(j\omega)\right|^2 d\omega + \frac{1}{2\pi}\int_{-\infty}^{\infty} S_\eta(\omega)\left|\Phi(j\omega)\right|^2 d\omega, \qquad (35)$$

где $\Phi_\varepsilon(j\omega)$ — частотная (амплитудно-фазовая) характеристика замкнутой системы по ошибке, определяемая по формуле (34), $\Phi(j\omega)$ — частотная характеристика замкнутой системы по входному воздействию:

$$\Phi(j\omega) = \frac{W(j\omega)}{1+W(j\omega)}. \qquad (36)$$

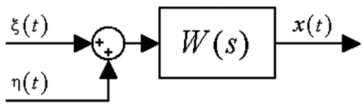


Рис. 2. Аддитивная смесь полезного сигнала и помехи на входе линейной системы

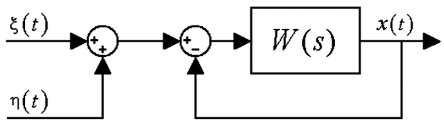


Рис. 3. Замкнутая система автоматического регулирования

Выражения (33), (35), определяющие дисперсию ошибки, представляют собой стандартные интегралы вида

$$J_n = \frac{1}{2\pi}\int\limits_{-\infty}^{\infty} \frac{h_n(j\omega)d\omega}{g_n(j\omega)g_n(-j\omega)} = \frac{(-1)^{n-1}N_n}{2a_0M_n}, \tag{37}$$

где $h_n(j\omega) = b_1(j\omega)^{2n-2} + b_2(j\omega)^{2n-4} + \ldots + b_n$;

$g_n(j\omega) = a_0(j\omega)^n + a_1(j\omega)^{n-1} + \ldots + a_n$.

Определители N_n и M_n составляют по определенным правилам из коэффициентов этих полиномов, например, для $n = 3$:

$$N_3 = \begin{vmatrix} b_1 & a_0 & 0 \\ b_2 & a_2 & a_1 \\ b_3 & 0 & a_3 \end{vmatrix}, \quad M_3 = \begin{vmatrix} a_1 & a_0 & 0 \\ a_3 & a_2 & a_1 \\ 0 & 0 & a_3 \end{vmatrix}. \tag{38}$$

Аналогично для $n = 4$:

$$N_4 = \begin{vmatrix} b_1 & a_0 & 0 & 0 \\ b_2 & a_2 & a_1 & a_0 \\ b_3 & a_4 & a_3 & a_2 \\ b_4 & 0 & 0 & a_4 \end{vmatrix}, \quad M_4 = \begin{vmatrix} a_1 & a_0 & 0 & 0 \\ a_3 & a_2 & a_1 & a_0 \\ 0 & a_4 & a_3 & a_2 \\ 0 & 0 & 0 & a_4 \end{vmatrix}. \tag{39}$$

Формулу (37) рекомендуется использовать для расчета дисперсии ошибки системы автоматического регулирования и в данной лабораторной работе.

1.3. Вычисление дисперсии сигнала на выходе линейной системы

Приведем пример расчета для разомкнутой системы, схема которой приведена на рис. 4.

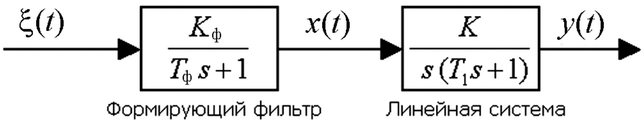


Рис. 4. Линейная система с формирующим фильтром

На вход системы подается случайный процесс — белый шум $\xi(t)$ со спектральной плотностью $S_\xi(\omega) = c^2$.

Сигнал на выходе формирующего фильтра с передаточной функцией $W_ф(s) = \dfrac{K_ф}{T_ф s + 1}$, т. е. входной сигнал линейной системы $x(t)$, будет иметь дисперсию

$$D_x = \frac{1}{2\pi}\int_{-\infty}^{\infty} c^2 \frac{K_ф^2}{\left|T_ф j\omega + 1\right|^2} d\omega. \tag{40}$$

Дисперсия выходного сигнала $y(t)$ будет равна

$$D_y = \frac{1}{2\pi}\int_{-\infty}^{\infty} c^2 \frac{K_ф^2}{\left|T_ф j\omega + 1\right|^2} \left|\Phi(j\omega)\right|^2 d\omega, \tag{41}$$

где

$$\left|\Phi(j\omega)\right|^2 = \frac{K^2}{\left|T_1(j\omega)^2 + j\omega + K\right|^2}.$$

Подынтегральные выражения в (40) и (41) представляют собой спектральные плотности соответствующих сигналов:

$$S_x(\omega)=\frac{c^2K_ф^2}{\left|T_ф j\omega+1\right|^2};$$
$$S_y(\omega)=\frac{c^2K_ф^2K^2}{\left|T_1T_ф(j\omega)^3+(T_1+T_ф)(j\omega)^2+(1+KT_ф)j\omega+K\right|^2}. \qquad (42)$$

Поскольку спектральные плотности (42) представляют собой дробно-рациональные функции, то вычисление интегралов (40), (41) сводится к вычислению стандартного интеграла (37).

Для дисперсии входного сигнала (40) получим:

$$h_1(j\omega)=c^2K_ф^2,\ g_1(j\omega)=T_ф j\omega+1.$$

Следовательно, коэффициенты многочленов

$$b_1=c^2K_ф^2;\ a_0=T_ф;\ a_1=1;$$
$$N_1=b_1=c^2K_ф^2;\ M_1=a_1=1.$$

Таким образом, по формуле (37) при $n=1$ находим:

$$D_x=\frac{(-1)^{1-1}N_1}{2a_0M_1}=\frac{c^2K_ф^2}{2T_ф}.$$

Проведем аналогичные расчеты для дисперсии выходного сигнала. В этом случае $n=3$,

$$h_3(j\omega)=0\cdot(j\omega)^4+0\cdot(j\omega)^2+c^2K_ф^2;$$

$$g_3(j\omega)=T_1T_ф(j\omega)^3+(T_1+T_ф)(j\omega)^2+(1+KT_ф)j\omega+K.$$

Выпишем коэффициенты многочленов:

$$b_1=0;\ b_2=0;\ b_3=c^2K_ф^2;$$
$$a_0=T_1T_ф;\ a_1=T_1+T_ф;\ a_2=1+KT_ф;\ a_3=K.$$

Теперь воспользуемся формулами (38):

$$N_3 = \begin{vmatrix} 0 & T_1T_ф & 0 \\ 0 & 1+KT_ф & T_1+T_ф \\ c^2K_ф^2 & 0 & K \end{vmatrix} = c^2K_ф^2T_1T_ф(T_1+T_ф);$$

$$M_3 = \begin{vmatrix} T_1+T_ф & T_1T_ф & 0 \\ K & 1+KT_ф & T_1+T_ф \\ 0 & 0 & K \end{vmatrix} = K(T_1+T_ф)(1+KT_ф) - KT_1T_ф.$$

Окончательно получим:

$$D_y = \frac{(-1)^{3-1}N_3}{2a_0M_3} = \frac{c^2K_ф^2T_1T_ф(T_1+T_ф)}{2T_1T_ф\left(K(T_1+T_ф)(1+KT_ф)-KT_1T\right)} =$$

$$= \frac{c^2K_ф^2(T_1+T_ф)}{2K\left((T_1+T_ф)(1+KT_ф)-T_1T\right)}.$$

Выше приведены основные математические формулы, используемые при выполнении лабораторных работ по статистическому анализу автоматических систем. Далее мы кратко остановимся на возможностях пакета MATLAB 7 при решении этих задач.

2. ПРИМЕНЕНИЕ ПАКЕТА MATLAB ДЛЯ СТАТИСТИЧЕСКОГО АНАЛИЗА АВТОМАТИЧЕСКИХ СИСТЕМ

MATLAB — это программный пакет для технических вычислений, объединяющий средства вычисления, визуализации и программирования. Этому пакету посвящено достаточно много пособий, с которыми читатель может ознакомиться самостоятельно (см., например, [3, 4]).

2.1. Определение характеристик эргодического случайного процесса с помощью пакета MATLAB

Ниже дано описание основных операций, выполняемых пакетом MATLAB при определении характеристик случайного процесса с использованием формул, приведенных в разд. 1.

Для определения среднего значения (19) по элементам вектора `values` используется функция

```
mean(values)
```

Она определяет среднее значение по каждому столбцу матрицы `values`.

Для расчета стандартного отклонения по формуле (25) применяется функция

```
s = std(values,1)
```

Стандартное отклонение рассчитывается по каждому столбцу матрицы `values`. Второй параметр (в нашем случае 1) указывает, что вычисление стандартного отклонения необходимо проводить именно в виде (25).

Для определения дисперсии случайного процесса (25) необходимо применить дополнительные преобразования:

```
disp = (std(values,1))^2/time_scale
```

где `time_scale` — период дискретизации значений `values`.

Для нахождения оценки корреляционной функции случайного процесса используется функция `xcov`:

```
[covv,lags] = xcov(values,'coeff')
```

Параметр `'coeff'` указывает MATLAB, что расчет выполняется по приведенной ниже формуле (43), следующей из формулы (24). Здесь $x_i = \xi(t_i)$ — измеренные значения реализации $\xi(t)$. Функция

$$c(m) = \frac{1}{N - m + 1} c_{xx}(m - N), \; m = 1, \ldots, 2N - 1, \tag{43}$$

где $c_{xx}(m) = \begin{cases} \sum\limits_{n=0}^{N-|m|-1} \left(x(n+m) - \frac{1}{N} \sum\limits_{i=0}^{N-1} x_i \right) \left(x(n) - \frac{1}{N} \sum\limits_{i=0}^{N-1} x_i \right) \text{ при } m \geq 0, \\ c_{xx}(-m) \text{ при } m < 0, \end{cases}$

определяет значения корреляционной функции `covv` для значений m, заданных вектором `lags`.

Для определения спектральной плотности случайного процесса по его корреляционной функции используется алгоритм быстрого дискретного преобразования Фурье — дискретный аналог формулы (11):

$$S_{\xi}(\omega) = \Delta t \sum_{i=1}^{N/2} K_{\xi}((i-1)\Delta t) \exp\left(\frac{-2\pi j}{N}(i-1)\omega\right). \qquad (44)$$

Этот алгоритм реализован функцией

```
Sp = abs(fft(covv))
```

Здесь `covv` — вектор значений корреляционной функции, `fft(covv)` — функция вызова алгоритма быстрого преобразования Фурье. Функция `abs()` используется для нахождения модуля комплексной величины.

2.2. Формирование случайного сигнала

Формирование случайного сигнала выполняется методом формирующего фильтра (см. подразд. 1.2). В связи с этим нам понадобятся функции, выполняемые пакетом MATLAB по формированию белого шума, стандартных сигналов, а также по формированию заданной передаточной функции формирующего фильтра.

Функция `randn` создает нормально распределенные случайные числа и массивы со средним значением 0, дисперсией $\sigma^2 = 1$. Эта функция имеет вид

`Y = randn(m,n)` или `Y = randn([m n])`

Она вычисляет матрицу случайных элементов размером $m \times n$.

Для изменения дисперсии случайного сигнала можно использовать следующую конструкцию:

```
for i = 1:size(Y)
        Y(i)= sqrt(disp_val)*Y(i);
end
```

смысл которой состоит в том, что значение случайного сигнала в каждый момент времени умножается на корень квадратный из желаемой дисперсии.

Для создания сигналов стандартного вида (синусоида, прямоугольные импульсы и т. д.) применяют функцию `gensig`:

```
[u,t] = gensig(type,tau,Tf,Ts)
```

Параметр `tau` задает период сигнала, `Tf` — длительность сигнала, `Ts` — дискретность задания сигнала в секундах. Параметр `type` определяет вид сигнала и может принимать следующие значения: `'sin'` — синусоида с периодом `tau`; `'square'` — периодический сигнал, равный нулю в течение первых `tau/2` секунд периода и единице в оставшееся время периода; `'pulse'` — периодический сигнал, равный нулю в течение первых `Ts` секунд периода и единице в оставшееся время периода.

Функция `gensig` определяет два вектора `u` и `t`. Компоненты вектора `t`: $t_i = T_s(i-1), \quad i = 1, 2, ..., \ (T_f / T_s + 1)$.

Компоненты вектора `u` – это значения сигнала в моменты времени t_i: $u_i = u(t_i)$. Например, команда

```
[u,t] = gensig('square',5,30,0.1)
```

создает периодические прямоугольные импульсы с периодом 5 с, длительностью 30 с, периодом дискретизации 0,1 с.

2.3. Моделирование линейной системы

Будем считать, что задана передаточная функция линейной системы. Требуется определить сигнал на выходе этой системы.

Для этого сначала необходимо записать передаточную функцию системы в виде, принятом в пакете MATLAB. Для этого используется функция `tf`. Поясним ее применение с помощью примера.

Пусть передаточная функция разомкнутой системы задана в виде

$$W_{\text{раз}}(s) = \frac{10000(5s+4)}{s^3 + 60s^2 + 500s + 10000}.$$

Требуется записать в коде программы эту передаточную функцию, а также передаточную функцию замкнутой системы. Запись имеет следующий вид:

```
s = tf('s')
% Передаточная функция разомкнутой системы
W_razomk = 10000*(5*s+4)/(s^3+...
              60*s^2+500*s+10000)
```

```
% Передаточная Функция замкнутой системы
W = W_razomk/(1+W_razomk)
```

В случае, если необходимо записать передаточную функцию линейного усилителя (например, $W(s) = 2$), необходимо использовать вырожденную форму записи:

```
s = tf('s')
% Передаточная функция
W = 2*s/s
```

В тексте программы знак `%` означает начало комментария — строка, начинающаяся с такого знака, исполняться не будет.

Для определения реакции системы на заданное воздействие $u(t)$ используется функция `lsim`:

```
lsim(sys,u,t)
```

Векторы `t` и `u` задаются аналогично функции `gensig` (см. выше). Матрица `u` должна иметь столько же столбцов, сколько входов в системе. Каждая строка `u(i,:)` определяет значения сигналов на входах системы в моменты времени `t(i)`.

3. МЕТОДИЧЕСКИЕ УКАЗАНИЯ К ВЫПОЛНЕНИЮ ЛАБОРАТОРНЫХ РАБОТ

3.1. Работа № 1. Формирование случайного сигнала с заданной спектральной плотностью

Цель работы — сформировать на ПЭВМ случайный процесс с заданной спектральной плотностью и провести исследование его характеристик.

Порядок выполнения работы

1. Для спектральной плотности случайного процесса, заданной преподавателем, сформировать передаточную функцию формирующего фильтра.
2. С помощью пакета MATLAB выполнить моделирование формирующего фильтра и убедиться в правильности составленной программы.

3. Сформировать случайный сигнал типа белого шума с заданной интенсивностью. Получить графическое отображение сигнала на экране монитора.

4. Определить статистические характеристики случайного сигнала на выходе формирующего фильтра (математического ожидания, спектральной функции, дисперсии, автокорреляционной функции) для десяти реализаций. Получить на экране вид реализаций случайного процесса на выходе формирующего фильтра.

5. Выполнить статистический анализ для ста реализаций. Сравнить полученные результаты с результатами обработки десяти реализаций. Распечатать графики корреляционной функции и спектральной плотности. Сравнить их с выражениями, которые были заданы. Сделать вывод о качестве выполненного эксперимента, т. е. о соответствии сформированного случайного сигнала заданному.

Для определения характеристик формирующего фильтра по заданной спектральной плотности случайного сигнала необходимо представить выражение для числителя и знаменателя спектральной плотности в следующем виде:

$$S(\omega) = \frac{P(j\omega)P(-j\omega)}{Q(j\omega)Q(-j\omega)},$$

где $P(j\omega)$ и $Q(j\omega)$ — многочлены от переменной $j\omega$.

Тогда с учетом (31) амплитудно-фазовая частотная характеристика формирующего фильтра будет иметь вид

$$W(j\omega) = \frac{P(j\omega)}{Q(j\omega)}.$$

Пример выполнения работы № 1

Спектральная плотность случайного сигнала задана в виде

$$S(\omega) = \frac{1}{0,25\omega^2 + 1} = \frac{1}{(-0,5j\omega + 1)(0,5j\omega + 1)}.$$

1. Определим частотную характеристику формирующего фильтра:

$$W(j\omega) = \frac{1}{0{,}5\,j\omega + 1}.$$

2. Сформируем с помощью пакета MATLAB требуемый случайный сигнал. Для этого вначале необходимо очистить пространство переменных от данных, которые могли остаться от предыдущих запусков программы:

```
clear;
s = tf('s')
```

3. Запишем передаточную функцию формирующего фильтра:

```
W_form = 1/(0.5*s+1)
```

4. Положим дисперсию белого шума равной единице:

```
disp_val=1
```

5. Будем обрабатывать одновременно 10 реализаций случайного процесса:

```
num_ser = 10
```

6. Зададим период дискретизации:

```
time_scale = 0.001
```

7. Зададим размеры массива и заполним массив времени. Длительность периода обработки установим равной 2 с. Если время переходного процесса в системе превышает 1 с (это станет понятно при выполнении лабораторной работы), то длительность периода обработки необходимо увеличить.

```
t_obrabotki = 2;
[u,t] = gensig('square', 1, t_obrabotki, ...
            time_scale);
size_u = size(u,1)
```

8. Создадим 10 реализаций белого шума с длиной `size_u` и дисперсией `disp_val`:

```
r = randn(num_ser,size(u,1));
noise_= zeros(num_ser,size(u,1));
for series = 1:num_ser
```

```
for i = 1:size(u,1)
    noise_(series,i) = sqrt(disp_val)* ...
            r(series,i);
end; end;
```

9. Пропустим семейство реализаций белого шума через формирующий фильтр при помощи функции `lsim`:

```
ai_noise = zeros(1,size(u,1));
modell_inp = zeros(num_ser,size(u,1));
for series = 1:num_ser
    for i = 1:size(u,1)
         ai_noise(i)=noise_(series,i);
    end;
    model_i_noise=lsim(W_form,ai_noise,t);
    for i = 1:size(u,1)
         modell_inp(series,i) = ...
                model_i_noise(i);
end; end;
```

10. Выполним оценку результатов. Исследуем для этого семейство реализаций выходного сигнала формирующего фильтра `modell_inp`. Для этого значения массива `modell_inp` запишем в массив `values`:

```
values= zeros(num_ser,size(u,1));
for j = 1:num_ser
        for i = 1:size(u,1)
             values(j, i) = modell_inp(j,i);
end; end;
```

11. Проведем расчет среднего по множеству реализаций случайного процесса по формуле (14):

```
mean_= mean(values,1);
```

12. Определим дисперсию случайного процесса по формуле (16) с коррекцией на шаг дискретизации:

```
std_= std(values,0,1);
disp_= zeros(1,size_u);
for i = 1:size_u
```

```
  disp_(i) = std_(i)^2/time_scale;
end;
```

13. Рассчитаем оценку корреляционной функции и спектральной плотности по формулам (15), (26), (43), (44):

```
value_r = zeros(1,size(u,1));
covv = zeros(num_ser, 2*size_u-1);
NFFT = 2^nextpow2(size_u);
spec = zeros(num_ser, NFFT);
for series = 1:num_ser
   for i = 1:size_u
        value_r(i) = values(series,i);
   end;
   [covvb,lags_] = xcov(value_r,'coeff');
   for i = 1:(2*size_u-1)
       covv(series,i) = covvb(i);
   end;
   temp_sp = 2*abs(fft(value_r,NFFT))/size_u;
   for i = 1:NFFT
       spec(series, i) = temp_sp(i);
   end;
end;
cov_= mean(covv,1);
f = 1/time_scale*0.5*linspace(-0.125, ...
0.125, NFFT/8);
spect_= mean(spec,1);
lags_= lags_*time_scale;
```

14. Отобразим результаты на экран в виде графиков.
14.1. Белый шум (рис. 5):

```
plot(t, noise_); grid on;
xlabel('t, секунды', 'FontSize', 12);
set(gca, 'FontSize', 12);
```

14.2. Белый шум, пропущенный через фильтр (рис. 6):

```
plot(t,values); grid on;
xlabel('t, секунды', 'FontSize', 12);
set(gca, 'FontSize', 12);
```

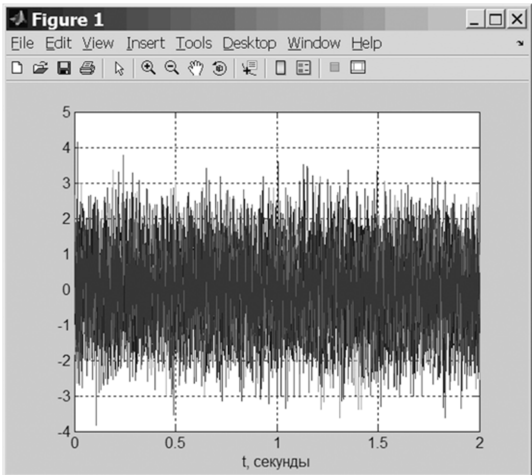


Рис. 5. Белый шум

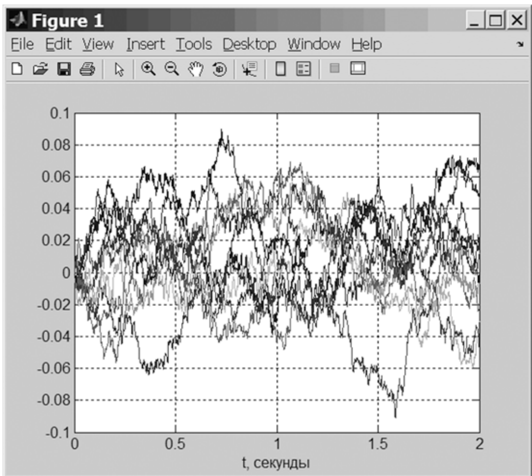


Рис. 6. Белый шум, пропущенный через фильтр

14.3. Среднее по множеству реализаций выходного сигнала — оценка математического ожидания (рис. 7):

```
plot (t, mean_); grid on;
xlabel('t, секунды', 'FontSize', 12);
set(gca, 'FontSize', 12);
```

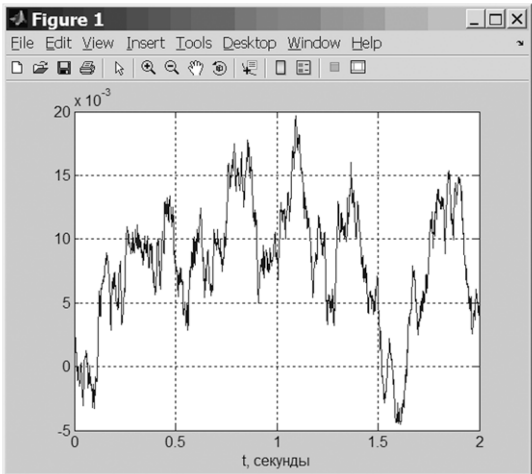


Рис. 7. Среднее по множеству реализаций выходного сигнала

14.4. Дисперсия (рис. 8):

```
plot (t, disp_); grid on;
xlabel('t, секунды', 'FontSize', 12);
set(gca, 'FontSize', 12);
```

14.5. Корреляционная функция (рис. 9):

```
plot (lags_, cov_); grid on;
xlabel('\tau, секунды', 'FontSize', 12);
ylabel('K(\tau)', 'FontSize', 12);
title('Корреляционная функция', ...
        'FontSize', 12);
set(gca, 'FontSize', 12);
```

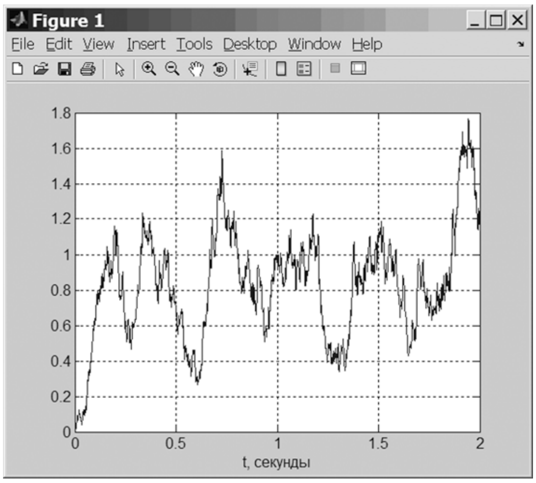


Рис. 8. Дисперсия выходного сигнала системы

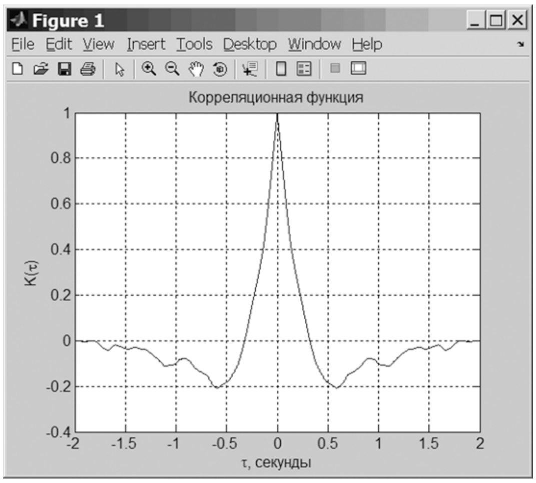


Рис. 9. Корреляционная функция выходного сигнала системы

14.6. Спектральная плотность (рис. 10):

```
plot(f, [spect_(NFFT/16:-1:1) ...
            spect_(1:NFFT/16)]); grid on;
xlabel('\omega, Гц', 'FontSize', 12);
```

```
ylabel('S(\omega)', 'FontSize', 12);
title('Спектральная плотность', ...
        'FontSize', 12);
set(gca, 'FontSize', 12);
```

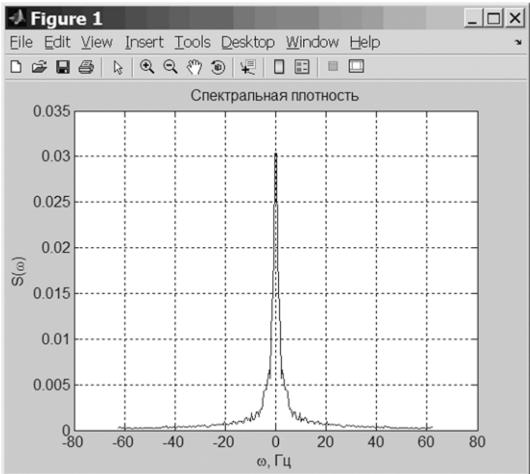


Рис. 10. Спектральная плотность выходного сигнала системы

Наблюдаемый провал графика в районе малых частот вызван тем, что частота обратно пропорциональна периоду колебаний, и для малых частот не хватает длительности периода наблюдения для построения достоверной оценки спектральной плотности.

Требования к отчету

Отчет о выполненной работе должен содержать:

– текст команд, выполненных в среде MATLAB,

– распечатки графиков, отображающих основные статистические характеристики процессов на выходе формирующего фильтра (математическое ожидание, дисперсию, корреляционную функцию и спектральную плотность);

– сопоставление характеристик заданного и полученного в результате эксперимента случайных процессов.

3.2. Работа № 2. Анализ линейной системы при воздействии случайного возмущения и детерминированного полезного сигнала

Цель работы — исследование характеристик случайного сигнала на выходе линейной системы, на вход которой подается сумма полезного ступенчатого сигнала и случайного процесса с заданными характеристиками.

Порядок выполнения работы

1. Для спектральной плотности случайного процесса, заданной преподавателем, сформировать передаточную функцию формирующего фильтра.
2. С помощью пакета MATLAB выполнить моделирование формирующего фильтра и убедиться в правильности составленной программы.
3. Сформировать случайный сигнал типа белого шума с заданной интенсивностью. Получить графическое отображение сигнала на экране монитора.
4. Определить статистические характеристики случайного сигнала на выходе линейной системы (математическое ожидание, спектральную функцию, дисперсию, автокорреляционную функцию) для десяти реализаций. Получить на экране вид реализаций случайного процесса на выходе линейной системы.
5. Выполнить статистический анализ для ста реализаций. Сравнить полученные результаты с результатами обработки десяти реализаций. Распечатать графики корреляционной функции и спектральной плотности.
6. Провести аналитическое исследование случайного сигнала на выходе системы и сравнить полученные результаты с экспериментальными результатами.

В этой лабораторной работе на вход линейной системы (рис. 11) подается полезный сигнал $x(t)$ в виде единичного ступенчатого воздействия и случайный процесс с заданной спектральной плотностью, рассматриваемый в качестве помехи. Ошибкой системы является разность между полезным сигналом и выходом системы: $\varepsilon(t) = y(t) - x(t)$. Целью работы является проведение моделирования и определение статистических характеристик сигнала $\varepsilon(t)$.

Пример выполнения работы № 2

Выполнение лабораторной работы рассмотрим на примере системы, представленной на рис. 11.

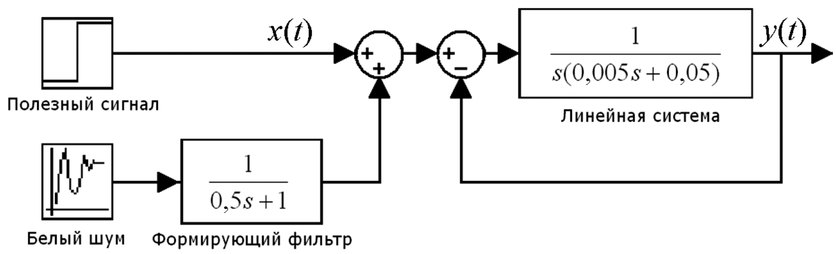


Рис. 11. Структурная схема системы. На входе сумма полезного детерминированного сигнала и случайной помехи

1. По методике, описанной в работе № 1, рассчитаем параметры формирующего фильтра. Создадим семейство реализаций помехи с заданной спектральной плотностью. Число реализаций `num_ser = 100`.

2. Сформируем полезный сигнал в виде единичного ступенчатого сигнала на интервале $t \in [0; 2]$. Предположим, что при $t \in [0; 0,2]$ его значение равно нулю, при $t \in (0,2; 2]$ — единице:

```
step_= zeros(1,size_u);
for i = 1:size(u,1)
      step_(i) = 0+(i>(.2/time_scale));
end;
```

3. Получим семейство реализаций входа системы:

```
input = zeros(num_ser, size(u,1));
for series = 1:num_ser
for i = 1:size(u,1)
 input(series,i) = modell_inp(series,i) ...
         + step_(i);
end;
end;
```

4. Выведем на экран входной сигнал системы (рис. 12):

```
plot(t, input); grid on;
xlabel('t, секунды', 'FontSize', 12);
set(gca, 'FontSize', 12);
```

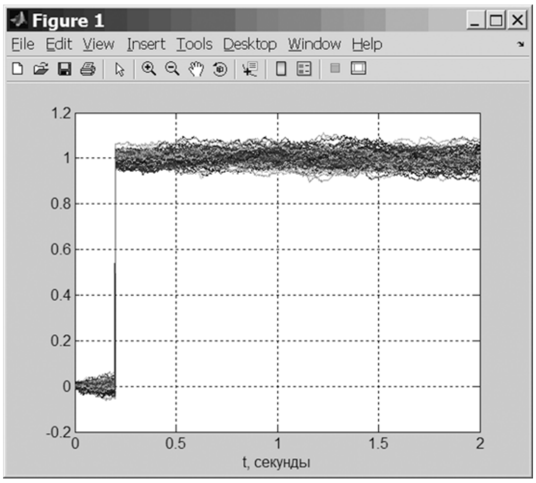


Рис. 12. Случайный сигнал на входе системы

5. Пропустим входной сигнал через линейную систему:

```
s = tf('s')
```

5.1. Введем передаточную функцию разомкнутой системы:

```
W_s_razomk = 1/(0.005*s^2+0.05*s)
```

5.2. Введем передаточную функцию замкнутой системы:

```
W_s = W_s_razomk/(1+ W_s_razomk)
```

5.3. Вычислим выходной сигнал системы:

```
model_out=zeros(num_ser, size(u,1));
input_cur=zeros(1, size(u,1));
for series = 1:num_ser
    for i = 1:size(u,1)
```

```
        input_cur(i)= input(series,i);
    end;
    model_output=lsim(W_s, input_cur,t);
    for i = 1:size(u,1)
        modell_out(series,i)= ...
           model_output(i);
    end;
end;
```

6. Выведем на экран выходной сигнал системы (рис. 13):

```
plot(t, modell_out); grid on;
xlabel('t, секунды', 'FontSize', 12);
set(gca, 'FontSize', 12);
```

7. Вычислим сигнал ошибки системы:

```
err_=zeros(num_ser, size(u,1));
for series = 1:num_ser
  for i = 1:size(u,1)
     err_(series,i) = modell_out(series,i)...
           - step_(i);
end; end;
```

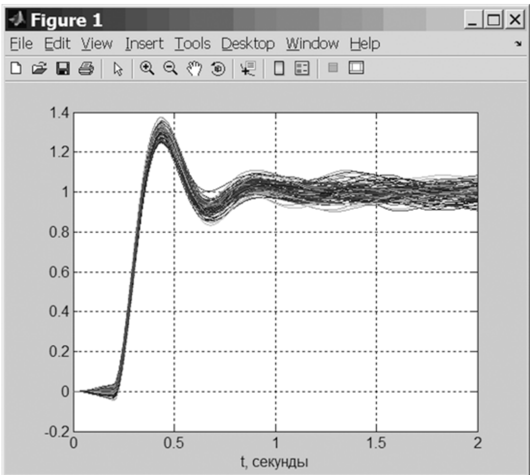


Рис. 13. Случайный сигнал на выходе системы

8. Выведем на печать сигнал ошибки системы (рис. 14):

```
plot (t, err_); grid on;
xlabel('t, секунды', 'FontSize', 12);
set(gca, 'FontSize', 12);
```

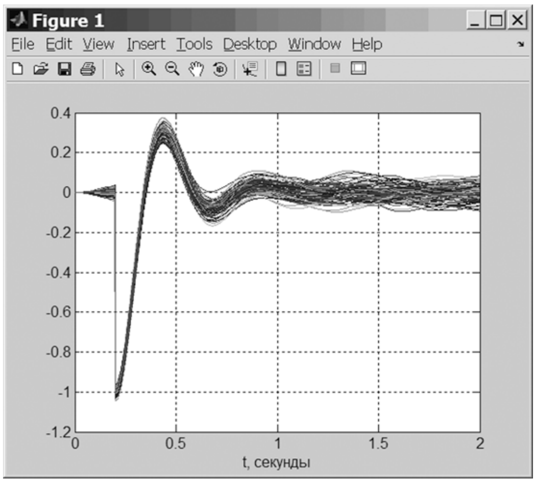


Рис. 14. Случайный сигнал ошибки системы

9. Видно, что случайный процесс для ошибки системы на интервале от 0 до 2 с нельзя считать стационарным. Тем не менее по алгоритмам, изложенным в описании работы № 1, определим среднее, дисперсию, корреляцию и спектральную плотность сигнала ошибки системы для нестационарного процесса.

9.1. Для анализа семейства реализаций сигнала ошибки системы значения массива `err_` запишем в массив `values`:

```
start_time = 0;
% Начало периода анализа наблюдений
values = zeros(num_ser, size(u,1)- ...
         start_time/time_scale);
for j = 1:num_ser
 for i = (start_time/time_scale+1):size(u,1)
     values(j,i-start_time/time_scale) = ...
          err_(j,i);
```

```
  end;
end;
```

9.2. Проведем расчет среднего по реализациям случайного процесса по формуле (14):

```
mean_= mean(values,1);
```

9.3. Выполним расчет дисперсии случайного процесса по формуле (16), с коррекцией на шаг дискретизации

```
std_= std(values,0,1);
disp_= zeros(1,size(values,2));
for i = 1:size(values,2);
     disp_(i) = std_(i)^2/time_scale;
end;
```

9.4. Рассчитаем оценку корреляционной функции и спектральной плотности по формулам (15), (26), (43), (44):

```
value_r = zeros(1, size(values,2));
covv = zeros(num_ser, 2*size(values,2)-1);
NFFT = 2^nextpow2(size(values,2));
spec = zeros(num_ser, NFFT);
for series = 1:num_ser
    for i = 1: size(values,2)
         value_r(i) = values(series,i);
    end;
    [covvb,lags_] = xcov(value_r,'coeff');
    for i = 1:(2* size(values,2)-1)
        covv(series,i) = covvb(i);
    end;
    temp_sp = 2*abs(fft(value_r,NFFT))/ ...
              size(values,2);
    for i = 1:NFFT
        spec(series, i)=temp_sp(i);
    end;
end;
cov_= mean(covv,1);
f = 1/time_scale*0.5*linspace(-0.125, ...
              0.125,NFFT/8);
spect_= mean(spec,1);
```

```
lags_= lags_*time_scale;
```

10. Отобразим результаты расчета на экране.
10.1. Среднее по реализациям ошибки системы (рис. 15):

```
plot(t((start_time/time_scale+1) ...
               :size(u,1)), mean_); grid on;
xlabel('t, секунды', 'FontSize', 12);
set(gca, 'FontSize', 12);
```

10.2. Дисперсия ошибки системы (рис. 16):

```
plot(t((start_time/time_scale+1) ...
            :size(u,1)), disp_); grid on;
xlabel('t, секунды', 'FontSize', 12);
set(gca, 'FontSize', 12);
```

10.3. Корреляционная функция ошибки системы (рис. 17):

```
plot(lags_, cov_); grid on;
xlabel('\tau, секунды', 'FontSize', 12);
ylabel('K(\tau)', 'FontSize', 12);
title('Корреляционная функция', ...
'FontSize', 12); set(gca, 'FontSize', 12);
```

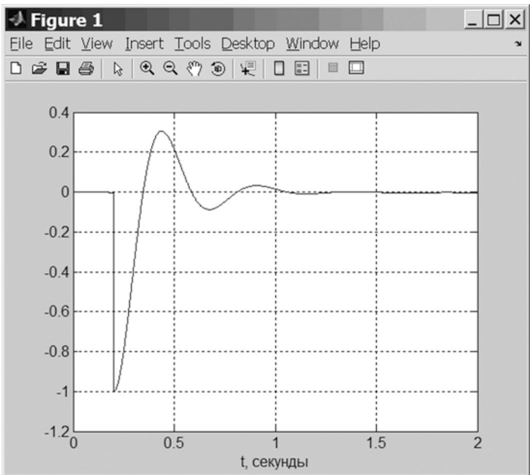


Рис. 15. Среднее по реализациям ошибки

Рис. 16. Дисперсия ошибки системы

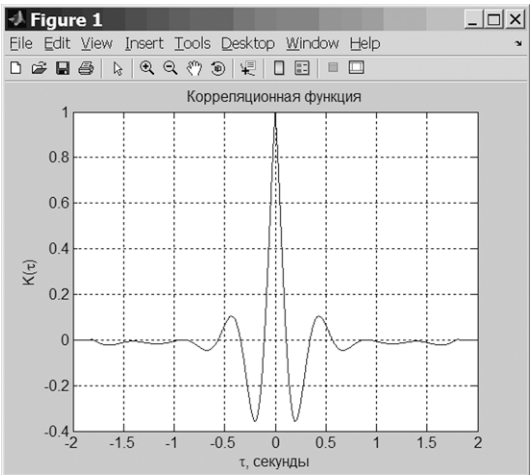


Рис. 17. Корреляционная функция ошибки системы

10.4. Спектральная плотность ошибки системы (рис. 18):

```
plot(f, [spect_(NFFT/16:-1:1) ...
               spect_(1:NFFT/16)]); grid on;
```

```
xlabel('\omega, Гц', 'FontSize', 12);
ylabel('S(\omega)', 'FontSize', 12);
title('Спектральная плотность', ...
'FontSize', 12); set(gca, 'FontSize', 12);
```

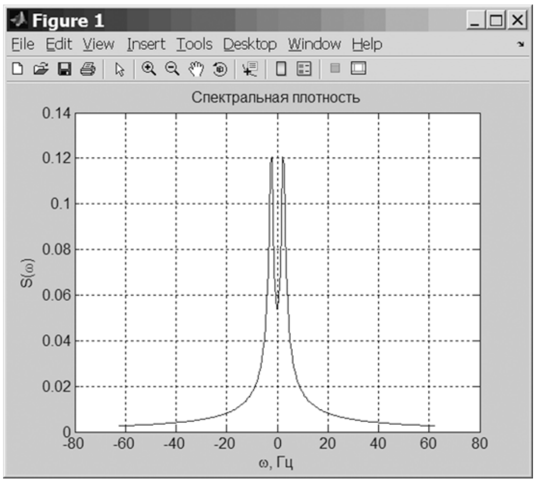


Рис. 18. Спектральная плотность ошибки системы

11. Теперь учтем, что определение параметров процесса необходимо проводить для стационарных процессов. По графику, показанному на рис. 14, определим длительность переходного процесса. Время моделирования (`t_obrabotki`) должно быть на порядок больше длительности переходного процесса. Анализ полученных результатов должен проводиться после затухания переходного процесса. Для нашего примера время затухания переходного процесса составляет примерно 0,8 с. Затухание переходного процесса происходит уже после первой секунды. Выберем `t_obrabotki=10; start_time=2` (с запасом). Результаты повторного выполнения этой работы для нового временного интервала показаны на рис. 19 – 22.

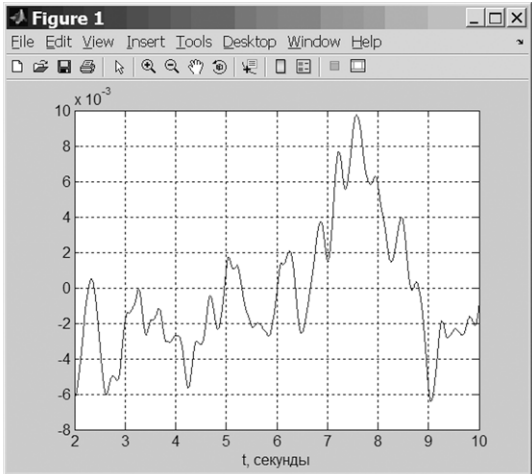


Рис. 19. Среднее по реализациям ошибки

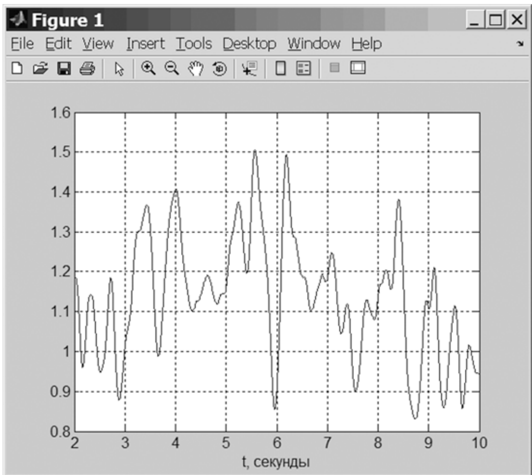


Рис. 20. Дисперсия ошибки системы

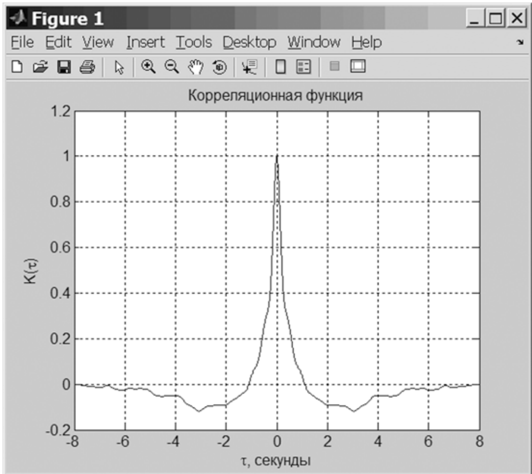


Рис. 21. Корреляционная функция ошибки системы

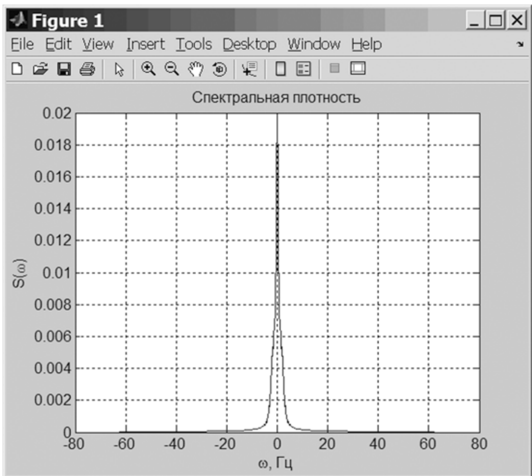


Рис. 22. Спектральная плотность ошибки системы

Требования к отчету

Отчет о выполненной работе должен содержать:

– текст команд, выполненных в среде MATLAB,

– распечатки графиков, отображающих основные статистические характеристики процессов на выходе линейной системы (математическое ожидание, дисперсию, корреляционную функцию и спектральную плотность);

– сопоставление корреляционной функции ошибки системы и спектральной плотности ошибки системы в случае выполнения расчетов по всему периоду наблюдений и по части наблюдений с завершившимся переходным процессом;

– сопоставление результатов эксперимента с результатами, полученными аналитическим путем.

3.3. Работа № 3. Анализ линейной системы при воздействии случайного полезного сигнала и случайного шума

Цель работы — исследовать линейную систему автоматического управления, на вход которой подается аддитивная смесь полезного случайного сигнала и случайной помехи (шума). Предварительное аналитическое исследование системы выполняется предварительно в форме домашнего задания. Результаты аналитического расчета проверяются экспериментально с помощью математического моделирования с использованием пакета MATLAB.

Порядок выполнения работы

1. Для случайных процессов, статистические характеристики которых указаны в домашнем задании, сформировать передаточные функции формирующих фильтров.

2. С помощью пакета MATLAB выполнить моделирование формирующих фильтров и убедиться в правильности составленной программы.

3. Сформировать случайные сигналы с заданными спектральными характеристиками. Получить графическое отображение сигналов на экране монитора.

4. Определить статистические характеристики случайного сигнала на выходе линейной системы (математическое ожидание, спектральную функцию, дисперсию, автокорреляционную функцию) для десяти реализаций. Вывести на экран вид реализаций случайного процесса на выходе линейной системы.

5. Провести статистический анализ для ста реализаций. Сравнить полученные результаты с результатами обработки десяти реализаций. Распечатать графики корреляционной функции и спектральной плотности.

6. Сравнить полученные экспериментально результаты с результатами аналитического расчета.

Пример выполнения работы № 3

Выполнение лабораторной работы рассмотрим на примере исследования системы, представленной на рис. 23.

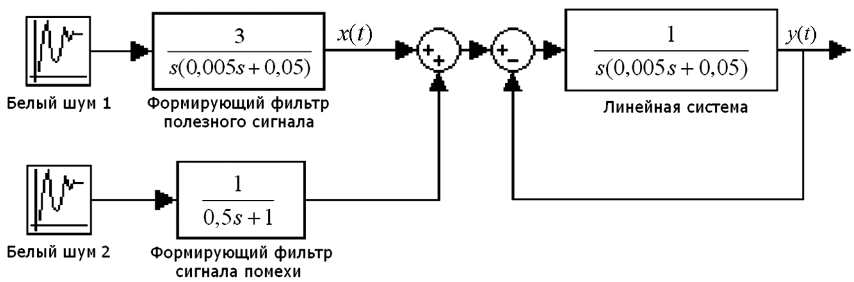


Рис. 23. Структурная схема системы. На вход системы поступает аддитивная смесь полезного случайного сигнала и помехи

Ошибкой системы в данном случае является разность между выходом системы и полезным сигналом: $\varepsilon(t) = y(t) - x(t)$ (см. рис. 23). Мерой этой ошибки является ее дисперсия.

Работу будем выполнять в следующем порядке.

1. По методике, описанной в работе № 1, рассчитаем параметры формирующего фильтра для шума с заданной спектральной плотностью.

2. По той же методике сформируем полезный сигнал с заданной спектральной плотностью.

Ниже показано, как в коде программы в MATLAB можно одновременно создать помеху и полезный сигнал.

Очистим вначале пространство переменных:

```
clear;
s = tf('s')
```

2.1. Запишем передаточную функцию формирующего фильтра для помехи:

```
W_form = 1/(0.5*s+1)
```

2.2. Запишем передаточную функцию формирующего фильтра для полезного сигнала:

```
W_p = 3/(0.005*s^2+0.05*s+1)
```

2.3. Положим дисперсию белого шума на входе формирующего фильтра помехи равной единице:

```
disp_val = 1
```

2.4. Аналогично для входа формирующего фильтра полезного сигнала:

```
disp_val_p = 1
```

3. Будем обрабатывать одновременно 100 реализаций случайного процесса:

```
num_ser = 100
```

4. Выберем период дискретизации:

```
time_scale = 0.001
```

5. Зададим размеры массива времени и заполним его. Длительность периода обработки сигнала была выбрана при выполнении работы № 2.

```
t_obrabotki = 10;
[u,t] = gensig('square', 1, ...
           t_obrabotki,time_scale);
size_u = size(u,1)
```

6. Создадим массив `num_ser` реализаций белого шума с длиной `size_u` и дисперсией `disp_val` и `disp_val_p`:

```
r = randn(num_ser,size(u,1));
r_p = randn(num_ser,size(u,1));
noise_= zeros(num_ser,size(u,1));
noise_p = zeros(num_ser,size(u,1));
for series = 1:num_ser
for i = 1:size(u,1)
   noise_(series,i) = sqrt(disp_val)*...
           r(series,i);
   noise_p(series,i) = sqrt(disp_val_p)*...
           r_p(series,i);
end;
end;
```

7. Пропустим семейство реализаций белого шума через формирующий фильтр при помощи функции `lsim`:

```
ai_noise = zeros(1,size(u,1));
ai_noise_p = zeros(1,size(u,1));
modell_inp = zeros(num_ser,size(u,1));
modell_inp_p = zeros(num_ser,size(u,1));
for series = 1:num_ser
    for i = 1:size(u,1)
         ai_noise(i) = noise_(series,i);
         ai_noise_p(i) = noise_p(series,i);
    end;
    model_i_noise = lsim(W_form,ai_noise,t);
    model_i_noise_p = lsim(W_p,ai_noise_p,t);
    for i = 1:size(u,1)
         modell_inp(series,i)= ...
            model_i_noise(i);
         modell_inp_p(series,i)= ...
            model_i_noise_p(i);
    end;
end;
```

8. Получим семейство реализаций случайного процесса на входе системы:

```
input = zeros(num_ser,size(u,1));
for series = 1:num_ser
for i = 1:size(u,1)
```

```
  input(series,i) = modell_inp(series,i)+ ...
            modell_inp_p(series,i);
end;
end;
```

9. Выведем на печать графики входного сигнала системы (для удобства сравнения графиков будем рассматривать лишь первые две секунды — рис. 24):

```
t_to_plot = 2;
plot(t(1:(t_to_plot/time_scale+1)), ...
    input(:,1:(t_to_plot/ time_scale+1)));
grid on;
xlabel('t, секунды', 'FontSize', 12);
set(gca, 'FontSize', 12);
```

10. Пропустим входной сигнал через систему, воспользовавшись алгоритмом, описанном в работе № 2.

10.1. Предварительно запишем передаточные функции разомкнутой и замкнутой системы:

```
W_s_razomk = 1/(0.005*s^2+0.05*s)
W_s = W_s_razomk/(1+ W_s_razomk)
```

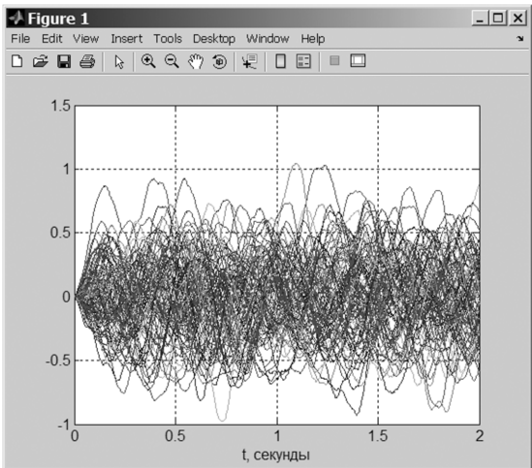


Рис. 24. Входной сигнал системы

10.2. Вычислим выходной сигнал:

```
input_cur = zeros(1,size(u,1));
modell_out = zeros(num_ser,size(u,1));
for series = 1:num_ser
    for i = 1:size(u,1)
         input_cur(i) = input(series,i);
    end;
    model_output = lsim(W_s, input_cur,t);
    for i = 1:size(u,1)
         modell_out(series,i) = ...
            model_output(i);
    end;
end;
```

10.3. Выведем выходной сигнал на экран (рис. 25):

```
plot(t(1:(t_to_plot/time_scale+1)), ...
modell_out(:,1:(t_to_plot/time_scale+1)));
grid on;
xlabel('t, секунды', 'FontSize', 12);
set(gca, 'FontSize', 12);
```

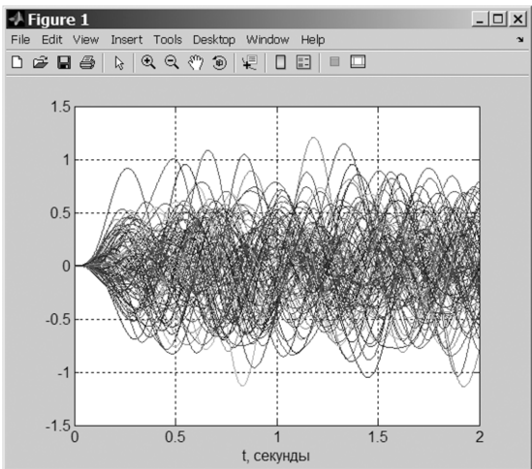


Рис. 25. Выходной сигнал системы

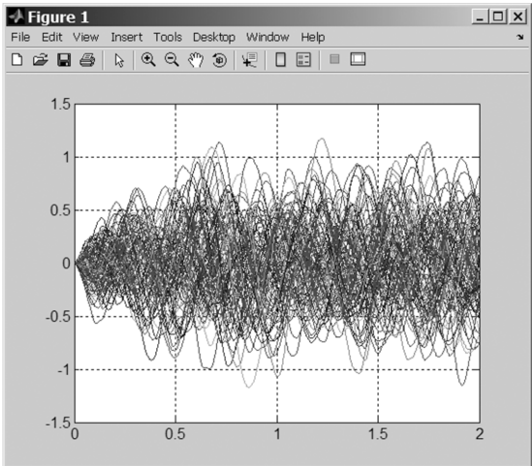


Рис. 26. Случайный сигнал ошибки системы

11. Определим ошибку системы и выведем ее на экран (рис. 26).

```
err_= zeros(num_ser,size(u,1));
for series = 1:num_ser
  for i = 1:size(u,1)
    err_(series,i) = modell_out(series,i)...
           -modell_inp_p(series,i);
  end;
end;
plot(t(1:(t_to_plot/time_scale+1)),  ...
    err_(:,1:(t_to_plot / time_scale+1)));
grid on; xlabel('t, секунды','FontSize',12);
set(gca, 'FontSize', 12);
```

12. Определим статистические характеристики выходного сигнала и ошибки системы (математическое ожидание, дисперсию, корреляционную функцию, спектральную плотность) по алгоритму, описанному в работе № 2.

13. Выполним оценку результатов. Исследуем семейство реализаций ошибки системы `err_` после затухания переходного процесса (время `start_time` примем таким же, как в работе № 2). Для этого значения массива `err_` запишем в массив `values`:

```
start_time = 2; %начало периода анализа
values = zeros(num_ser, size(u,1)- ...
                 start_time/time_scale);
for j = 1:num_ser
 for i = (start_time/time_scale+1):size(u,1)
   values(j, i- start_time/time_scale)...
                 = err_(j,i);
 end;
end;
```

14. Рассчитаем среднее по множеству реализаций случайного процесса по формуле (14):

```
mean_ = mean(values,1);
```

15. Вычислим дисперсию случайного процесса по формуле (16) с коррекцией на шаг дискретизации:

```
std_= std(values,0,1);
disp_= zeros(1,size(values,2));
for i = 1:size(values,2);
   disp_(i) = std_(i)^2/time_scale;
end;
```

16. Определим оценку корреляционной функции и спектральной плотности по формулам (15), (26), (43), (44):

```
value_r = zeros(1, size(values,2));
covv  =zeros(num_ser, 2*size(values,2)-1);
NFFT = 2^nextpow2(size(values,2));
spec = zeros(num_ser, NFFT);
for series = 1:num_ser
    for i = 1: size(values,2)
         value_r(i) = values(series,i);
    end;
    [covvb,lags_] = xcov(value_r,'coeff');
    for i = 1:(2* size(values,2)-1)
        covv(series,i)=covvb(i);
    end;
    temp_sp = 2*abs(fft(value_r,NFFT)) ...
             / size(values,2);
```

```
    for i = 1:NFFT
        spec(series, i) = temp_sp(i);
    end;
end;
cov_ = mean(covv,1);
f = 1/time_scale*0.5*linspace(-0.125, ...
       0.125,NFFT/8);
spect_= mean(spec,1);
lags_= lags_*time_scale;
```

17. Отобразим результаты на экране монитора.
17.1. Среднее по реализациям ошибки (рис. 27):

```
plot(t((start_time/time_scale+1): ...
       size(u,1)), mean_); grid on;
xlabel('t, секунды', 'FontSize', 12);
set(gca, 'FontSize', 12);
```

17.2. Дисперсия ошибки (рис. 28):

```
plot(t((start_time/time_scale+1): ...
       size(u,1)), disp_); grid on;
xlabel('t, секунды', 'FontSize', 12);
set(gca, 'FontSize', 12);
```

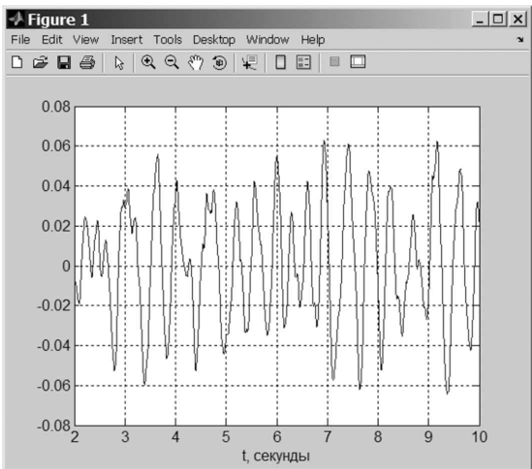


Рис. 27. Среднее по реализациям ошибки

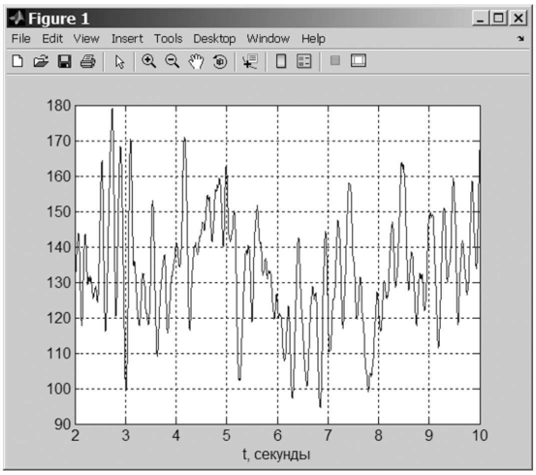


Рис. 28. Дисперсия ошибки системы

17.3. Корреляционная функция (рис. 29):

```
plot(lags_, cov_); grid on;
xlabel('\tau, секунды', 'FontSize', 12);
ylabel('K(\tau)', 'FontSize', 12);
title('Корреляционная функция', ...
            'FontSize', 12);
set(gca, 'FontSize', 12);
```

17.4. Спектральная плотность (рис. 30):

```
plot (f, [spect_(NFFT/16:-1:1) ...
         spect_(1:NFFT/16)]); grid on;
xlabel('\omega, Гц', 'FontSize', 12);
ylabel('S(\omega)', 'FontSize', 12);
title('Спектральная плотность', ...
      'FontSize', 12);
set(gca, 'FontSize', 12);
```

Рис. 29. Корреляционная функция ошибки системы

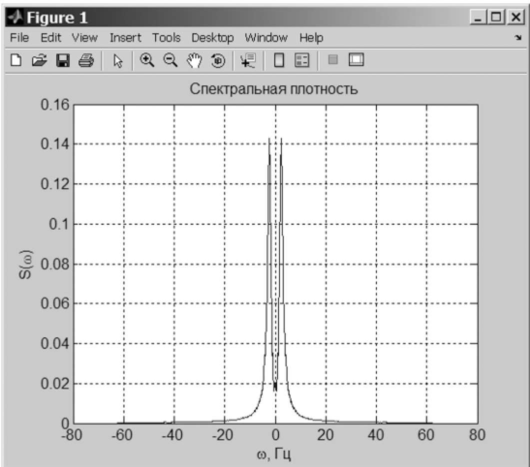


Рис. 30. Спектральная плотность ошибки системы

Требования к отчету

Отчет о выполненной работе должен содержать:

– текст команд, выполненных в среде MATLAB;

– распечатки графиков, отображающих основные статистические характеристики процессов (выходной сигнал линейной системы, ее ошибку, математическое ожидание, дисперсию, корреляционную функцию, спектральную плотность);

– сопоставление результатов, полученных в лабораторной работе, с результатами предварительно выполненных аналитических расчетов.

СПИСОК ЛИТЕРАТУРЫ

1. *Астапов Ю.М., Медведев В.С.* Статистическая теория систем автоматического регулирования и управления. М.: Наука, 1982. 304 с.

2. Методы классической и современной теории автоматического управления. В 5 т. Т. 1: Анализ и статистическая динамика систем автоматического управления / К.А. Пупков, Н.Д. Егупов, А.И. Баркин и др. М.: Изд-во МГТУ им. Н.Э. Баумана, 2000. 748 с.

3. *Дьяконов В.П.* MATLAB 6.5 SP1/7.0 + Simulink 5/6 в математике и моделировании. М.: Солон-Пресс, 2005. 576 с

4. *Потемкин В.Г.* Вычисления в среде MATLAB. М.: Диалог-МИФИ, 2004. 720 с.

ОГЛАВЛЕНИЕ