

**А.И. Жученко
Н.А. Кубрак
И.М. Голинко**

**ДИНАМИКА ОБЪЕКТОВ С
СОСРЕДОТОЧЕННЫМИ
ПАРАМЕТРАМИ**

**Киев
НУТУ “КП”
2006**

Рецензенты: *А.П. Ладанюк*, д-р техн., проф.
(Национальный университет пищевых технологий)
Ю.О. Скрипник, д-р техн., проф.
(Киевский национальный университет технологии и дизайна)
Ю.К. Тодорцев, д-р техн., проф.
(Одесский национальный политехнический университет)

Ж94

Жученко А.И., Кубрак Н.А., Голинко И.М. Динамика объектов с сосредоточенными параметрами: Учебн. пособие. –К.: НТУУ«КПИ», 2006. -151с

Рассматривается использование преобразования Лапласа для исследования линейных аналоговых автоматических систем. На примере элементарных динамических звеньев производится расчет переходных и частотных характеристик элементов систем. Выводятся и анализируются аналитические модели типовых технологических объектов, рассматриваемых как элементы систем с сосредоточенными параметрами.

Приводятся алгоритмы пересчета одних способов задания динамических характеристик в другие. Эти алгоритмы реализуются в виде подпрограмм и демонстрационно-отладочных программ на Турбо Паскале.

Для студентов направления подготовки “Автоматизация и компьютерно-интегрированные технологии” для всех форм обучения, в том числе для студентов-иностранцев данной специальности.

Введение

Исследование автоматических систем на инженерном уровне, как правило, предполагает их математическое моделирование в виде систем обыкновенных дифференциальных уравнений. Тем самым отдельные элементы системы и система в целом рассматривается как динамический объект с сосредоточенными параметрами. Источником для построения такой модели могут быть либо аналитические исследования либо результаты специально поставленных экспериментов (переходные, импульсные или частотные характеристики). Методы пассивного эксперимента в данном пособии не рассматриваются. Математической базой, обеспечивающей исследование динамического поведения систем автоматизации является преобразование Лапласа, которое позволяет легко пересчитать различные формы представления динамических характеристик одни в другие в зависимости от целей и методов исследования.

Возможность классифицировать и упорядочить бесконечное разнообразие элементов автоматических систем (собственно объектов управления, датчиков, преобразователей, устройств дистанционной передачи, регуляторов, исполнительных механизмов, регулирующих органов и т.д.) реализуется в форме отнесения элементов систем к тем или иным элементарным динамическим звеньям или их комбинациям. Формирование набора таких “элементарных” звеньев и рассмотрение их динамических характеристик (чаще всего переходных и/или частотных характеристик) позволяет в дальнейшем исследователю легко идентифицировать элементы по их динамическим свойствам (“узнавать их в лицо”).

Естественно, сколько-нибудь серьезное исследование динамики сложных автоматических систем не мыслится без компьютера. Поэтому в пособии рассматриваются компьютерные алгоритмы вычисления переходных и частотных характеристик исходя из самых различных форм задания динамических свойств объекта (системы обыкновенных дифференциальных уравнений, передаточной функции, массива ординат импульсной характеристики, частотных

характеристик). Расчет коэффициентов начального отрезка ряда Маклорена, в который разлагается передаточная функция исследуемого динамического канала, объясняется намерением в дальнейшем использовать этот ряд для идентификации объекта (нахождения аппроксимирующей передаточной функции, если точная передаточная функция либо неизвестна, либо чем-то не устраивает исследователя). Алгоритмы идентификации – это отдельная большая тема, выходящая за рамки настоящего пособия. Но подготовить некоторый задел в этом направлении будет не лишним.

В Приложении приводятся листинги программ, которые демонстрируют в работе алгоритмы, рассмотренные в тексте пособия. Эти программы могут быть использованы для организации цикла лабораторных занятий по рассматриваемой тематике.

Раздел 1. Преобразование Лапласа и его использование при исследовании автоматических систем

Современные методы исследования динамических систем и, в частности, систем автоматического управления так или иначе связаны с использованием преобразования Лапласа. Это преобразование позволяет легко перебрасывать мостик между переходными и частотными характеристиками, приводить в порядок и формализовать алгоритмы анализа и синтеза системы, упрощает разработку алгоритмов компьютерного моделирования автоматических систем.

Поэтому прежде чем приступить к рассмотрению компьютерных моделей и методов их исследования следует рассмотреть идею и простейшие свойства преобразования Лапласа. Читателю, у которого есть желание более основательно разобраться в этом мощном математическом инструменте, рекомендуем, например, такие работы [3, 7, 14].

1.1. Определение преобразования Лапласа и границы его применения

Пусть имеем некоторую функцию вещественной переменной $f(t)$, которую в дальнейшем будем называть оригиналом. В пособиях по математике в качестве независимой переменной, как правило, принято использовать x , но относительно специфики исследования динамических систем, где речь пойдет в первую очередь об изучении переходных процессов, то есть функций времени t , уместно рассматривать именно время как независимую переменную. Поэтому, мы и отдаем предпочтение переменной t .

Интеграл вида:

$$L[f(t)] = \int_0^{\infty} f(t) \cdot e^{-pt} dt, \quad (1.1)$$

носит название интеграла Лапласа. Здесь p - некоторая комплексная переменная.

Поскольку интегрирование выполняется по времени t , то после вычисления интеграла переменная t исчезает, вместе с тем появляется переменная p . Итак, в результате функции $f(t)$ вещественной переменной t ставится в соответствие $L[f(t)]$ – функция комплексной переменной p . Чтобы подчеркнуть это обстоятельство, результат преобразования (1.1) – изображение по Лапласу функции $f(t)$ мы будем записывать как $\bar{f}(p)$.

$$\bar{f}(p) = L[f(t)]. \quad (1.2)$$

Здесь черточка над обозначением функции напоминает читателю, что речь не идет о простой “механической” замене переменной t на p , а рассматривается именно преобразование Лапласа согласно формуле (1.1).

Функция $\bar{f}(p)$ называется изображением функции $f(t)$ по Лапласу. Как будет показано ниже, переход от оригинала к его изображению может значительно упрощать целый ряд операций над исследуемой функцией. Но не исключено, что наконец возникнет потребность обратного перехода, то есть возвращения к оригиналу. Чтобы этот переход был однозначным, надо как-то устранить возможную неоднозначность (точнее, неопределенность) поведения

функции $f(t)$ при $t < 0$, ведь согласно формуле (1.1) поведение функции при $t < 0$ никак не отображается на $\bar{f}(p)$. Функции, которые будут совпадать в диапазоне $0 < t < \infty$, но отличаться как угодно при $t < 0$, будут иметь абсолютно одинаковые изображения! Поэтому было принято радикальное решение: считать все функции-оригиналы, для которых мы собираемся выполнять преобразование (1.1), при $t < 0$ тождественно равными нулю

$$f(t)|_{t < 0} \equiv 0. \quad (1.3)$$

Это ход в стиле Александра Македонского, который вместо того чтобы распутывать Гордиев узел, его просто разрубил. Если у читателя еще остаются сомнения в корректности предложенного решения, обратим его внимание на особенности отсчета времени t . В своей практической деятельности люди не ведут отсчет времени с момента сотворения мира. Секундомер запускают (выставляют $t=0$) в момент стартового выстрела или в любой момент, признанный целесообразным пользователем (“С понедельника возьмусь”). Когда мы будем тестировать спроектированную систему, то мы будем подавать на ее вход определенные возмущения тогда, когда будем считать нужным. А до тех пор, пока это возмущение не появилось, его естественно считать равным нулю. Больше того, нуль отсчета практически любого физического параметра является условным, будь это температура, уровень, направление движения, высота и т.д. В таком случае кто вам мешает выбрать значение исследуемого параметра к началу процесса за ноль? Именно так и принято поступать в автоматике. Отсчет параметров ведут от их заданных значений. Так что проблем с выполнением условия (1.3) у нас не предвидится. Еще одно существенное условие: интеграл (1.1) должен быть конечным! Почему? Вообразите себе, что мы имеем ряд разных оригиналов, изображения которых равняются бесконечности. Но бесконечности нельзя отличить одну от другой! Итак, однозначный обратный переход становится невозможным. Из бесконечности нельзя возвратиться... Условие конечности (1.1) сводится к такому:

$$|f(t)|_{0 \leq t \leq \infty} \leq M \cdot e^{st}, 0 \leq t \leq \infty, \quad (1.4)$$

где M и s – некоторые положительные конечные числа.

Кстати, минимальное s , при котором (1.4) еще выполняется, называется абсциссой абсолютной сходимости.

На практике с выполнением условия (1.4) проблем не возникает. При $f(t) \rightarrow \infty$ любая математическая модель физического явления, процесса, аппарата – перестает быть адекватной. Здесь, как говорят, уже не до преобразования Лапласа ...

И в конце концов – последнее. Функция $f(t)$ должна быть непрерывной или, по крайней мере, кусочно-непрерывной в диапазоне $0 \leq t \leq \infty$. Она также должна иметь непрерывные производные. Применительно ко времени, как к дискретному параметру преобразование (1.1) не рассчитано. Для этого существует специальная модификация этого изображения.

1.2. Изображения элементарных функций

Чтобы у читателя сложилось хотя бы какое-то начальное впечатление об особенностях преобразования Лапласа, найдем изображения нескольких элементарных функций. Выбор этих функций, как может догадаться читатель, далеко не случайный. Нам придется с ними часто иметь дело.

Начнем с так называемой единичной ступенчатой функции $f(t) = 1(t)$, которая определяется следующим образом

$$1(t) = \begin{cases} 0 & \text{при } t < 0, \\ 1 & \text{при } t \geq 0. \end{cases} \quad (1.5)$$

График $1(t)$ представлен на рис. 1.1.

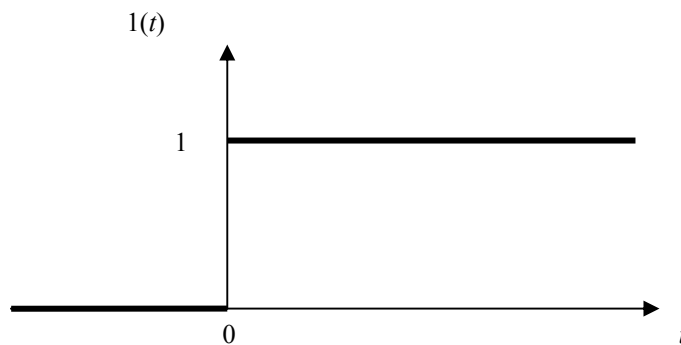


Рис. 1.1. График $1(t)$.

Кстати, выполнение условия (1.3) формально можно обеспечить, умножая $f(t)$ на $1(t)$. На самом деле

$$f(t) \cdot 1(t) \Big|_{t < 0} \equiv 0. \quad (1.6)$$

Особенностью (1.6) мы в дальнейшем часто будем пользоваться. Настолько часто, что как правило, даже не будем показывать в записи множитель $1(t)$, а будем просто считать его незримо присутствующим (по умолчанию).

В соответствии с (1.1)

$$L[1(t)] = \int_0^{\infty} 1(t) \cdot e^{-pt} dt.$$

Но в диапазоне $0 \leq t \leq \infty$ $1(t) \equiv 1$, итак, $L[1(t)] = \int_0^{\infty} e^{-pt} dt = -\frac{1}{p} \cdot e^{-pt} \Big|_0^{\infty} = \frac{1}{p}$.

Итак, окончательно имеем

$$L[1(t)] = \frac{1}{p}. \quad (1.7)$$

Пусть, теперь $f(t) = e^{-at}$ (множитель $1(t)$ не показан).

$$L[e^{-at}] = \int_0^{\infty} e^{-at} e^{-pt} dt = \int_0^{\infty} e^{-(p+a)t} dt = -\frac{1}{p+a} \cdot e^{-(p+a)t} \Big|_0^{\infty} = \frac{1}{p+a}$$

или

$$L[e^{-at}] = \frac{1}{p+a}. \quad (1.8)$$

Легко видеть, что (1.7) можно рассматривать как частный случай (1.8), а именно, при $a = 0$.

Пусть теперь $f(t) = t$ (точнее $t \cdot 1(t)$).

$$L[t] = \int_0^{\infty} t \cdot e^{-pt} dt.$$

Будем брать этот интеграл по частям: $u = t$, $du = dt$,

$$dv = e^{-pt} dt, \quad v = -\frac{1}{p} \cdot e^{-pt}.$$

Итак,

$$L[t] = -\frac{t}{p} \cdot e^{-pt} \Big|_0^{\infty} + \frac{1}{p} \int_0^{\infty} e^{-pt} dt.$$

Неопределенность в слагаемом $-\frac{t}{p} \cdot e^{-pt} \Big|_{t \rightarrow \infty}$ легко раскрывается по правилу Лопиталья (если снести e^{-pt} в знаменатель как e^{pt}).

$$L[t] = \frac{1}{p^2}. \quad (1.9)$$

Легко показать, что

$$L[t^2] = \frac{2}{p^3}. \quad (1.10)$$

В общем случае

$$L[t^k] = \frac{k!}{p^{k+1}}. \quad (1.11)$$

Читатель собственноручно может получить также формулу

$$L[t^k \cdot e^{-at}] = \frac{k!}{(p+a)^{k+1}}, \quad (1.12)$$

которая обобщает все ранее выведенные формулы.

А теперь возвратимся к формуле (1.8), положив в ней $a = \alpha + j\beta$

$$L[e^{-(\alpha+j\beta)t}] = \frac{1}{p + \alpha + j\beta}.$$

Проведем очевидное преобразование

$$L[e^{-\alpha t} \cdot e^{-j\beta t}] = \frac{p + \alpha - j\beta}{(p + \alpha)^2 + \beta^2},$$

$$L[e^{-\alpha t} \cos(\beta t) - je^{-\alpha t} \sin(\beta t)] = \frac{p + \alpha}{(p + \alpha)^2 + \beta^2} - j \frac{\beta}{(p + \alpha)^2 + \beta^2}.$$

Известно, что интеграл от суммы равняется сумме интегралов от слагаемых, последнее комплексное равенство распадается на два независимых равенства (вещественных и мнимых частей), то есть

$$L[e^{-\alpha t} \cos(\beta t)] = \frac{p + \alpha}{(p + \alpha)^2 + \beta^2}, \quad (1.13)$$

$$L[e^{-\alpha t} \sin(\beta t)] = \frac{\beta}{(p + \alpha)^2 + \beta^2}. \quad (1.14)$$

При $\alpha = 0$ формулы (1.13) и (1.14) дают

$$L[\cos(\beta t)] = \frac{p}{p^2 + \beta^2}, \quad (1.15)$$

$$L[\sin(\beta t)] = \frac{\beta}{p^2 + \beta^2}. \quad (1.16)$$

Для начала ограничимся приведенными соответствиями между оригиналами и изображениями.

1.3. Основные свойства преобразования Лапласа

Как видно из заголовка параграфа, мы собираемся рассмотреть лишь несколько самых простых свойств, которые имеют существенное значение именно для нашего предмета исследования.

Линейность преобразования Лапласа

Интеграл от суммы равняется сумме интегралов, а постоянный множитель можно выносить за знак интеграла:

$$L\left[\sum_{s=1}^n R_s \cdot f_s(t)\right] = \sum_{s=1}^n R_s \cdot \bar{f}_s(p). \quad (1.17)$$

Изменение масштаба

В интеграле (1.1)

$$\bar{f}(p) = \int_0^{\infty} f(t) \cdot e^{-pt} dt$$

выполним такие преобразования:

$$\bar{f}(p) = \int_0^{\infty} f\left(\frac{at}{a}\right) e^{-p \frac{at}{a}} d\left(\frac{at}{a}\right).$$

Введем обозначение $at=t_1$, $p/a=p_1$. Получим:

$$\bar{f}(ap_1) = \frac{1}{a} \int_0^{\infty} f\left(\frac{t_1}{a}\right) e^{-p_1 \cdot t_1} dt_1.$$

Опускаем индексы:

$$a \bar{f}(ap_1) = \int_0^{\infty} f\left(\frac{t}{a}\right) \cdot e^{-pt} dt. \quad (1.18)$$

Таким образом, делению вещественной переменной на a соответствует умножение комплексной переменной и самого изображения на a .

Сдвиг в вещественной области

В интеграле (1.1)

$$\bar{f}(p) = \int_0^{\infty} f(t) \cdot e^{-pt} dt$$

выполняем замену $t=t_1-\tau$, где $\tau \geq 0$.

Тогда
$$\bar{f}(p) = \int_{\tau}^{\infty} f(t_1 - \tau) \cdot e^{-p(t_1 - \tau)} dt.$$

Умножив левую и правую части полученного равенства на $e^{-p\tau}$, имеем:

$$e^{-p\tau} \cdot \bar{f}(p) = \int_{\tau}^{\infty} f(t_1 - \tau) \cdot e^{-pt_1} dt.$$

Если $f(t_1 - \tau) = 0$ при $0 \leq t \leq \tau$, то нижнюю границу интегрирования в последнем выражении можно заменить нулем. Тогда, опуская индекс, получим:

$$e^{-p\tau} \cdot \bar{f}(p) = L[f(t - \tau)]. \quad (1.19)$$

Таким образом, сдвиг вещественной переменной t в области оригиналов на величину τ вправо отвечает умножению изображения на $e^{-p\tau}$.

Сдвиг в комплексной области

В формуле (1.1)

$$\bar{f}(p) = \int_0^{\infty} f(t) \cdot e^{-pt} dt$$

выполним замену переменной $p = p_1 + a$

$$\bar{f}(p_1 + a) = \int_0^{\infty} f(t) \cdot e^{-(p_1 + a)t} dt$$

или, что то же

$$\bar{f}(p_1 + a) = \int_0^{\infty} (f(t) \cdot e^{-at}) \cdot e^{-p_1 t} dt.$$

Опуская индекс при p_1 получаем:

$$\bar{f}(p + a) = L(e^{-at} f(t)). \quad (1.20)$$

Таким образом, умножению оригинала на e^{-at} отвечает сдвиг аргумента изображения на a .

Наглядным примером только что сформулированного свойства могут служить пары формул (1.7)-(1.8), (1.11-1.12).

Дифференцирование в комплексной области

В формуле (1.1) выполним дифференцирование по p :

$$\frac{d}{dp}[\bar{f}(p)] = \int_0^{\infty} f(t) \cdot (-t) \cdot e^{-pt} dt,$$

откуда

$$-\frac{d}{dp}[\bar{f}(p)] = L[t \cdot f(t)]. \quad (1.21)$$

Таким образом, умножению оригинала на t отвечает дифференцирование изображения по p с умножением результата на -1 .

Формулу (1.21) можно, например, применить чтобы перейти от формулы (1.7) к (1.8), от (1.9) к (1.10).

Интегрирование в комплексной области

Интегрируем по p в пределах от p до ∞ выражение (1.1):

$$\int_p^{\infty} \bar{f}(p) dp = \int_p^{\infty} \int_0^{\infty} f(t) \cdot e^{-pt} dt dp.$$

Изменяем порядок интегрирования правой части:

$$\int_p^{\infty} \bar{f}(p) dp = \int_0^{\infty} f(t) \cdot \left(\int_p^{\infty} e^{-pt} dp \right) dt.$$

Возьмем внутренний интеграл

$$\int_p^{\infty} \bar{f}(p) dp = \int_0^{\infty} \frac{f(t)}{t} e^{-pt} dt,$$

или

$$\int_p^{\infty} \bar{f}(p) dp = L\left[\frac{f(t)}{t}\right]. \quad (1.22)$$

Итак, делению оригинала на t отвечает интегрирование изображения в пределах от p до ∞ .

Читатель может применить формулу (1.22) для перехода от (1.9) к (1.7), (1.10) к (1.9).

Дифференцирование в области оригинала

Возьмем интеграл (1.1)

$$\bar{f}(p) = \int_0^{\infty} f(t) \cdot e^{-pt} dt.$$

Интеграл в правой части будем брать по частям. Для этого вводим обозначения.

$$\begin{aligned} u &= f(t), & du &= f'(t)dt, \\ dv &= e^{-pt} dt, & v &= -\frac{1}{p} e^{-pt}. \end{aligned}$$

Имеем

$$\bar{f}(p) = \frac{-f(t)}{p} e^{-pt} \Big|_0^{\infty} + \frac{1}{p} \int_0^{\infty} f'(t) \cdot e^{-pt} dt.$$

При $t = \infty$ первое слагаемое, учитывая что $f(t)|_{t \rightarrow \infty} \neq \infty$, будет нулем, а $f(t)|_{t \rightarrow 0}$ будет пределом, к которому приближается $f(t)$ справа от нуля.

Итак,

$$L[f'(t)] = p\bar{f}(p) - f(0_+). \quad (1.23)$$

Если взять по частям интеграл

$$L[f'(t)] = \int_0^{\infty} f'(t) e^{-pt} dt,$$

будем иметь

$$L[f''(t)] = p^2 \bar{f}(p) - pf(0_+) - f'(0_+). \quad (1.24)$$

Можно показать, что

$$L[f^{(k)}(t)] = p^k \bar{f}(p) - \sum_{s=0}^{k-1} p^s f^{(k-s-1)}(0_+). \quad (1.25)$$

Если начальные условия нулевые, то формула (1.25) упрощается

$$L[f^{(k)}(t)] = p^k \bar{f}(p). \quad (1.26)$$

Таким образом, операция дифференцирования в области оригиналов сводится к элементарным алгебраическим операциям в области изображений, а при нулевых начальных условиях – к элементарному умножению на p в степени, которая равняется порядку производной.

Интегрирование в области вещественной переменной

Возьмем интеграл (1.1) по частям, считая что

$$u = e^{-pt}, \quad du = -p \cdot e^{-pt} dt,$$

$$dv = f(t)dt, \quad v = \int_0^t f(t)dt.$$

Будем иметь

$$\bar{f}(p) = e^{-pt} \int_0^t f(t)dt \Big|_0^\infty + p \cdot \int_0^\infty \left(\int_0^t f(t)dt \right) \cdot e^{-pt} dt,$$

откуда

$$\frac{\bar{f}(p)}{p} = L\left(\int_0^t f(t)dt\right), \quad (1.27)$$

Итак, интегрированию оригинала в пределах от 0 до t соответствует деление изображения на p .

Читатель может потренироваться в применении формулы (1.27) для выявления связи между формулами (1.7), (1.9)-(1.11).

Предельные значения оригинала

Согласно (1.23)

$$\int_0^\infty f'(t)e^{-pt} dt = p\bar{f}(p) - f(0_+).$$

Рассмотрим это соотношение при $p \rightarrow \infty$, откуда

$$\lim_{t \rightarrow 0} f(t) = \lim_{p \rightarrow \infty} p \cdot \bar{f}(p) \quad (1.28)$$

Рассмотрим теперь (1.23) при $p \rightarrow 0$,

$$f(t)_0^\infty = \lim_{p \rightarrow 0} [p \cdot \bar{f}(p) - f(0_+)],$$

откуда

$$\lim_{t \rightarrow \infty} f(t) = \lim_{p \rightarrow 0} p \cdot \bar{f}(p). \quad (1.29)$$

Формулы (1.28) и (1.29) определяют предельные значения оригинала. Из них, в частности, следует, что поведение оригинала $f(t)$ при малых значениях t определяется поведением изображения при больших значениях p и наоборот.

1.4. Передаточные функции, переходные, импульсные и частотные характеристики

Рассмотрим динамическую систему, которая описывается, например, таким дифференциальным уравнением

$$a_2 y'' + a_1 y' + a_0 y = b_1 x' + b_0 x. \quad (1.30)$$

Здесь x -вход, y -выход системы.

Можно было бы, конечно, один из ненулевых коэффициентов этого уравнения сделать равным единице, но мы оставим его в таком виде как есть для обобщения. Мы также собираемся потом обобщить результат на произвольные порядки левой и правой части уравнения.

Выполним над (1.30) преобразование Лапласа (поэлементно)

$$a_2 [p^2 \bar{y}(p) - py_0 - y'_0] + a_1 [p\bar{y}(p) - y_0] + a_0 \bar{y}(p) = b_1 [p\bar{x}(p) - x_0] + b_0 \bar{x}(p).$$

Здесь $\bar{y}(p)$ и $\bar{x}(p)$ - изображения по Лапласу соответственно от выхода и входа, y_0, y'_0, x_0 - начальные условия.

Решим теперь последнее уравнение относительно $\bar{y}(p)$

$$\bar{y}(p) = \frac{b_1 p + b_0}{a_2 p^2 + a_1 p + a_0} \bar{x}(p) + \frac{a_2 (py_0 - y'_0) + a_1 y_0 - b_1 x_0}{a_2 p^2 + a_1 p + a_0}. \quad (1.31)$$

Как видно из (1.31) изображение решения уравнения представляется в виде двух составляющих, а именно: первой, которая зависит от входного сигнала (мы в дальнейшем будем называть ее вынужденной составляющей) и второй – от начальных условий (это будет соответственно свободная составляющая движения системы).

Полином от p в числителе свободной составляющей будем называть полиномом начальных условий $V_{n,y}(p)$. При нулевых начальных условиях этот полином, очевидно, будет равняться нулю, итак, свободная составляющая будет отсутствовать.

При таких условиях (нулевых начальных) изображение решения (1.30) выглядит так

$$\bar{y}(p) = W(p) \cdot \bar{x}(p), \quad (1.32)$$

где $W(p) = \frac{b_1 p + b_0}{a_2 p^2 + a_1 p + a_0}$ называют передаточной функцией системы.

Если (1.32) переписать

$$W(p) = \frac{\bar{y}(p)}{\bar{x}(p)}, \quad (1.33)$$

то передаточную функцию можно определить как отношение изображения выходного сигнала системы (канала, объекта, регулятора, и т.п.) к изображению входного сигнала при нулевых начальных условиях.

Для объектов с сосредоточенными параметрами передаточные функции имеют в общем случае вид отношения полиномов, или как говорят, являются дробно-рациональными

$$W(p) = \frac{b_m p^m + b_{m-1} p^{m-1} + \dots + b_1 p + b_0}{a_n p^n + a_{n-1} p^{n-1} + \dots + a_1 p + a_0}. \quad (1.34)$$

Для того, чтобы согласно (1.29), выходной сигнал системы был конечными при $t \rightarrow \infty$ как реакция на конечный входной сигнал, необходимо чтобы

$$m \leq n. \quad (1.35)$$

Условие (1.35) называется условием физической реализуемости системы. Передаточные функции объектов с распределенными параметрами являются трансцендентными, в чем мы будем иметь возможность убедиться позже.

Присущая объектам с сосредоточенными параметрами связь между дифференциальным уравнением, которое описывает некоторый динамический канал, и соответствующей ему передаточной функцией, является тривиальной, а именно левая часть дифференциального уравнения по очевидной схеме формирует знаменатель передаточной функции, тогда как правая часть – аналогичным образом определяет числитель.

Знаменатель передаточной функции с точностью до обозначения переменной совпадает с так называемым характеристическим полиномом системы. В дальнейшем мы не будем себя утруждать выбором различных буквенных символов для обозначения преобразования Лапласа и формирования

характеристических полиномов – ограничимся переменной p .

Для того, чтобы преобразовать характеристический полином в характеристическое уравнение достаточно приравнять его нулю. Таким приемом мы в дальнейшем будем пользоваться без дополнительных комментариев.

Отмечая простоту и однозначность перехода от обыкновенного дифференциального уравнения, которое целиком и полностью определяет динамические свойства исследуемого канала, к передаточной функции и наоборот, легко согласиться с тем, что дифференциальное уравнение и передаточная функция является, в сущности, разными формами представления одних и тех же динамических свойств канала. Но варианты задания этих свойств рассмотренными способами не исчерпываются.

Тренеру, чтобы оценить потенциальные возможности спортсмена, достаточно посмотреть его на тренировке, а лучше на дистанции или в игре (увидеть его в деле). Так вот, динамические свойства объекта также можно оценить по его реакции на один из тестовых входных сигналов. Тестов можно придумать великое множество. Рассмотрим несколько простейших.

Как один из них можно взять уже упоминавшийся единичный ступенчатый сигнал $1(t)$, изображение его согласно (1.7) равняется $\frac{1}{p}$. Если на вход объекта подается $x(t) = 1(t)$, то при нулевых начальных условиях реакцию $y(t)$ на него называют переходной характеристикой и обозначают ее как $h(t)$. С учетом (1.32) и (1.7) можно записать

$$\bar{h}(p) = \frac{W(p)}{p}. \quad (1.36)$$

Как второй возможный входной сигнал можно взять так называемый единичный импульс

$$\delta(t) = \begin{cases} 0 & \text{при } t \neq 0, \\ \infty & \text{при } t = 0, \end{cases} \quad (1.37)$$

но так, что $\int_{-\infty}^{\infty} \delta(t) dt = 1$.

Единичный импульс можно вообразить себе как очень узкий (кратковременный) но значительный по амплитуде импульс, форма которого несущественна, но, что важно, площадь которого равняется единице. Площадь характеризует энергию, которую несет в себе импульс (что-то наподобие удара, а энергия характеризует силу удара). Понятно, что единицу измерения энергии пользователь может выбирать по своему усмотрению.

Если попробовать дифференцировать по времени функцию (1.5), то становится очевидным, что

$$\frac{d}{dt}[1(t)] = \delta(t). \quad (1.38)$$

А теперь если учесть (1.7) и (1.23), выходит, что

$$L[\delta(t)] = 1. \quad (1.39)$$

Реакция объекта на единичный импульсный входной сигнал (при нулевых начальных условиях) называется импульсной характеристикой или, как иногда говорят, функцией веса. С учетом сказанного, а также (1.32) и (1.39) вытекает очевидная зависимость

$$\bar{g}(p) = W(p), \quad (1.40)$$

где $g(t)$ - импульсная характеристика, а $\bar{g}(p)$ - ее изображение по Лапласу.

Формула (1.40) объясняет почему иногда передаточную функцию определяют как изображение по Лапласу от импульсной характеристики.

Если принять за входной сигнал линейно нарастающий (рамповый) сигнал $t \cdot 1(t)$, то реакцию на такой вход (при нулевых начальных условиях, понятно) называют рамповой характеристикой. Изображение по Лапласу от $t \cdot 1(t)$ согласно (1.9) равняется $\frac{1}{p^2}$, тогда согласно (1.32)

$$\bar{r}(p) = \frac{W(p)}{p^2}. \quad (1.41)$$

Здесь $r(t)$ - рамповая характеристика, а $\bar{r}(p)$ - ее изображение по Лапласу.

Сравнивая (1.36), (1.40) и (1.41), легко убедиться в справедливости таких соотношений

Таблица 1. Соотношения между переходными, импульсными и рамповыми характеристиками

В области изображений	В области оригиналов
$\bar{g}(p) = p\bar{h}(p)$	$g(t) = \frac{d}{dt}[h(t)]$
$\bar{h}(p) = p\bar{r}(p)$	$h(t) = \frac{d}{dt}[r(t)]$
$\bar{g}(p) = p^2\bar{r}(p)$	$g(t) = \frac{d^2}{dt^2}[r(t)]$
$\bar{h}(p) = \frac{\bar{g}(p)}{p}$	$h(t) = \int_0^t g(t)dt$
$\bar{r}(p) = \frac{\bar{h}(p)}{p}$	$r(t) = \int_0^t h(t)dt$
$\bar{r}(p) = \frac{\bar{g}(p)}{p^2}$	$r(t) = \int_0^t \int_0^t g(t)dt^2$

И, в конце концов, очень коротко о частотных характеристиках. Если на вход системы подать гармонический сигнал, например, $x = A_{\text{вх}} \sin(\omega t)$, где $A_{\text{вх}}$ - его амплитуда, а ω – угловая частота, то после затухания переходных составляющих, на выходе устойчивой системы (об устойчивости – ниже) установятся колебания типа $y = A_{\text{вых}} \sin(\omega t + \psi)$, где $A_{\text{вых}}$ - их амплитуда, ψ - сдвиг по фазе.

Если описанный „эксперимент” повторить при разных значениях частоты ω , то окажется, что для линейной системы отношение амплитуд и сдвиг по фазе будут функциями частоты (для нелинейных систем они кроме этого еще и будут зависеть от $A_{\text{вх}}$). Итак

$$\frac{A_{\text{вых}}}{A_{\text{вх}}} = A(\omega), \quad (1.42)$$

$$\psi = \varphi(\omega). \quad (1.43)$$

Функции $A(\omega)$ и $\varphi(\omega)$ называются амплитудно-частотной и фазо-частотной

характеристиками системы.

Если воспользоваться полярной системой координат и значение амплитудно-частотной характеристики рассматривать как модуль, фазо-частотной характеристики как аргумент, а значения частоты как параметр, то совокупность точек, которые соответствуют множеству значений параметра ω , образуют так называемую амплитудно-фазовую характеристику. Если теперь на полярную систему наложить комплексную плоскость так, чтобы нулевому значению φ соответствовала вещественная ось, то появляется возможность рассматривать амплитудно-фазовую характеристику как функцию от частоты ω . Можно показать, что амплитудно-фазовую характеристику (ее еще называют комплексной частотной характеристикой) можно получать из передаточной функции простой заменой параметра p на $j\omega$, откуда и ее обозначения

$$W(j\omega) = W(p)|_{p=j\omega}. \quad (1.44)$$

И тогда с учетом (1.42) и (1.43)

$$A(\omega) = \text{mod}[W(j\omega)], \quad (1.45)$$

$$\varphi(\omega) = \text{arg}[W(j\omega)]. \quad (1.46)$$

С другой стороны, вектор $W(j\omega)$ можно спроектировать на вещественную и мнимую оси

$$\text{Re}(\omega) = \text{re}[W(j\omega)], \quad (1.47)$$

$$\text{Im}(\omega) = \text{im}[W(j\omega)], \quad (1.48)$$

$\text{Re}(\omega)$ - вещественно-частотная, $\text{Im}(\omega)$ - мнимо-частотная характеристики.

Зависимость между названными характеристиками легко отслеживать, имея перед глазами такой элементарный прямоугольный треугольник

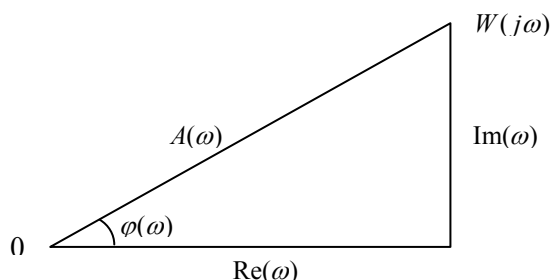


Рис 1.2. Связь между частотными характеристиками

Анализ рис. 1.2 приводит к очевидным соотношениям.

$$A(\omega) = \sqrt{\operatorname{Re}^2(\omega) + \operatorname{Im}^2(\omega)}, \quad (1.49)$$

$$\varphi(\omega) = \operatorname{arctg} \frac{\operatorname{Im}(\omega)}{\operatorname{Re}(\omega)}, \quad (1.50)$$

$$\operatorname{Re}(\omega) = A(\omega) \cos[\varphi(\omega)], \quad (1.51)$$

$$\operatorname{Im}(\omega) = A(\omega) \sin[\varphi(\omega)]. \quad (1.52)$$

Поскольку частотные характеристики отображают те же динамические свойства объекта, что и характеристики, показанные в таблице 1, то, очевидно существует возможность их пересчета одних в другие. Мы возвратимся к этой проблеме.

1.5. Элементарные динамические звенья – их дифференциальные уравнения и передаточные функции

Элементы динамических систем, очевидно, следует классифицировать по их динамическим свойствам. Ну, скажем, как классифицируют спортсменов – спринтеры, прыгуны, боксеры... Динамические свойства описываются, например, дифференциальными уравнениями. Так что за основу классификации можно взять тип дифференциального уравнения.

Рассмотрим в качестве элементарных такой набор типов дифференциальных уравнений. Будем называть их элементарными динамическими звеньями.

Безынерционное звено

$$y = kx. \quad (1.53)$$

Здесь и дальше x - вход, y -выход объекта.

Интегральное звено

$$y = k \cdot \int_0^t x dt. \quad (1.54)$$

Идеальное дифференциальное звено

$$y = kx'. \quad (1.55)$$

Реальное дифференциальное звено

$$Ty' + y = kTx'. \quad (1.56)$$

Интегро-дифференциальное звено

$$Ty' + y = k(\alpha Tx' + x). \quad (1.57)$$

Апериодическое звено I порядка

$$Ty' + y = kx. \quad (1.58)$$

Апериодическое звено II порядка

$$a_2 y'' + a_1 y' + y = kx. \quad (1.59)$$

при условии, что $a_1^2 - 4a_2 \geq 0$.

Колебательное звено

$$a_2 y'' + a_1 y' + y = kx \quad (1.60)$$

при условии, что $a_1^2 - 4a_2 < 0$.

Консервативное звено

$$a_2 y'' + y = kx. \quad (1.61)$$

Звено транспортного запаздывания

$$y(t) = x(t - \tau). \quad (1.62)$$

Приведенный комплект “элементарных звеньев” можно рассматривать как своего рода “джентльменский набор”. Слесарь-сантехник, слесарь-электрик, участковый врач – каждый из них ходит по вызовам со своим чемоданчиком, в котором находится минимальный набор необходимых инструментов. Для специалиста по автоматизации надо иметь собственный набор в первую очередь необходимых “звеньев”. Следует отметить, что каждый уважающий себя специалист формирует себе собственный набор, который может отличаться от типового. На выбор могут влиять и особенности конкретного “рабочего места” даже в пределах одной профессии. Так, что читатель может при необходимости расширить или сузить набор “элементарных” звеньев. Но иметь представление о возможностях и особенностях перечисленных звеньев все-таки необходимо.

Одно замечание относительно выше перечисленного набора звеньев. При формировании набора мы включили лишь звенья, в дифференциальных уравнениях которых все коэффициенты положительны. Такой подход не является общепринятым, как и вообще нет критерия, по которому звенья следует относить к элементарным. Понятие элементарности относительное. При желании можно было бы считать элементарными, например, только безынерционное и интегральное звенья, остальные можно “сконструировать” из этих двух и их комбинаций.

Физически такой параметр как k (коэффициент передачи) может быть как отрицательным, так и положительным (как говорят “fifty-fifty”), но фактически знак лучше (удобнее) отображать на сумматорах, с помощью которых реализуется замыкание систем. Мы так и будем делать.

Переход от обыкновенного дифференциального уравнения к передаточной функции мы уже рассматривали. Итак, приведем передаточные функции элементарных динамических звеньев без комментариев.

Безынерционное звено

$$W(p) = k. \quad (1.63)$$

Интегральное звено

$$W(p) = \frac{k}{p}. \quad (1.64)$$

Идеальное дифференциальное звено

$$W(p) = kp. \quad (1.65)$$

Реальное дифференциальное звено

$$W(p) = k \frac{Tp}{Tp + 1}. \quad (1.66)$$

Интегро-дифференциальное звено

$$W(p) = k \frac{\alpha Tp + 1}{Tp + 1}. \quad (1.67)$$

Апериодическое звено I порядка

$$W(p) = \frac{k}{Tp + 1}. \quad (1.68)$$

Апериодическое звено II порядка и колебательное звено

$$W(p) = \frac{k}{a_2 p^2 + a_1 p + 1}. \quad (1.69)$$

Консервативное звено

$$W(p) = \frac{k}{a_2 p^2 + 1}. \quad (1.70)$$

Звено транспортного запаздывания

$$W(p) = e^{-p\tau}. \quad (1.71)$$

Последняя передаточная функция получена с использованием теоремы сдвига в вещественной области (1.19).

В отличие от первых девяти элементарных звеньев, которые описывают объекты с сосредоточенными параметрами и передаточные функции которых являются дробно-рациональными, звено транспортного запаздывания служит элементарным примером объекта с распределенными параметрами.

Передаточные функции объектов с распределенными параметрами определяются из соответствующих дифференциальных уравнений или их систем (часть из которых является дифференциальными уравнениями в частных производных) не так просто, как это делалось в предыдущих случаях. Общей особенностью передаточных функций объектов с распределенными параметрами является то, что они являются трансцендентными.

1.6. Переходные характеристики элементарных динамических звеньев

Напоминаем, что переходная характеристика $h(t)$ – это реакция на входной сигнал $x = 1(t)$ при нулевых начальных условиях.

Безынерционное звено согласно (1.53)

$$h(t) = k \cdot 1(t). \quad (1.72)$$

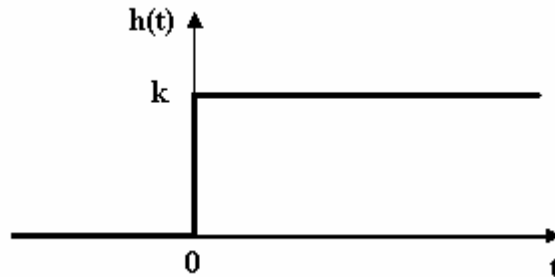


Рис. 1.3. Переходная характеристика безынерционного звена

Интегральное звено по формуле (1.54)

$$h(t) = kt \cdot 1(t). \quad (1.73)$$

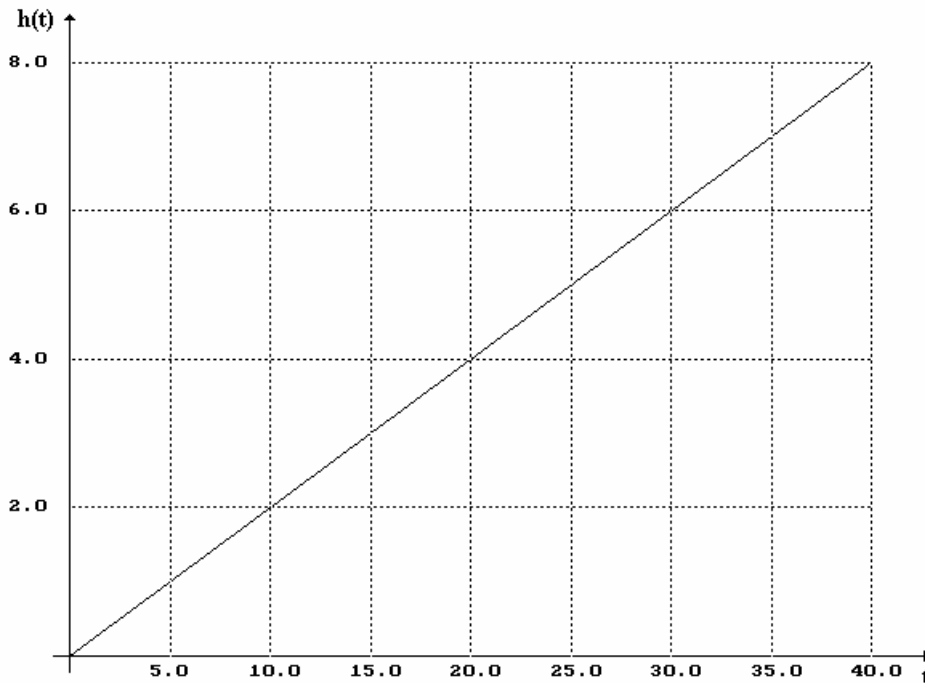


Рис. 1.4. Переходная характеристика интегрального звена, $W(p) = 0.2/p$.

Идеальное дифференциальное звено. Учитывая (1.38) и (1.55) имеем

$$h(t) = k \cdot \delta(t). \quad (1.74)$$

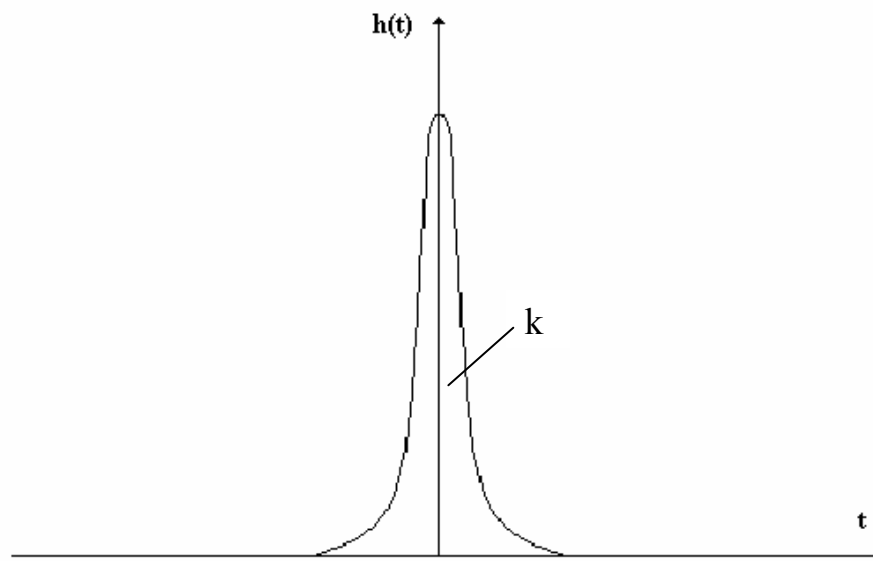


Рис. 1.5. Переходная характеристика идеального дифференциального звена

Реальное дифференциальное звено. Воспользуемся формулами (1.36) и (1.66). Получим

$$\bar{h}(p) = \frac{kT}{Tp + 1} = \frac{k}{p + \frac{1}{T}}.$$

Для выполнения обратного преобразования воспользуемся зависимостью (1.8) с учетом свойства линейности (1.17)

$$h(t) = ke^{-\frac{t}{T}}. \quad (1.75)$$

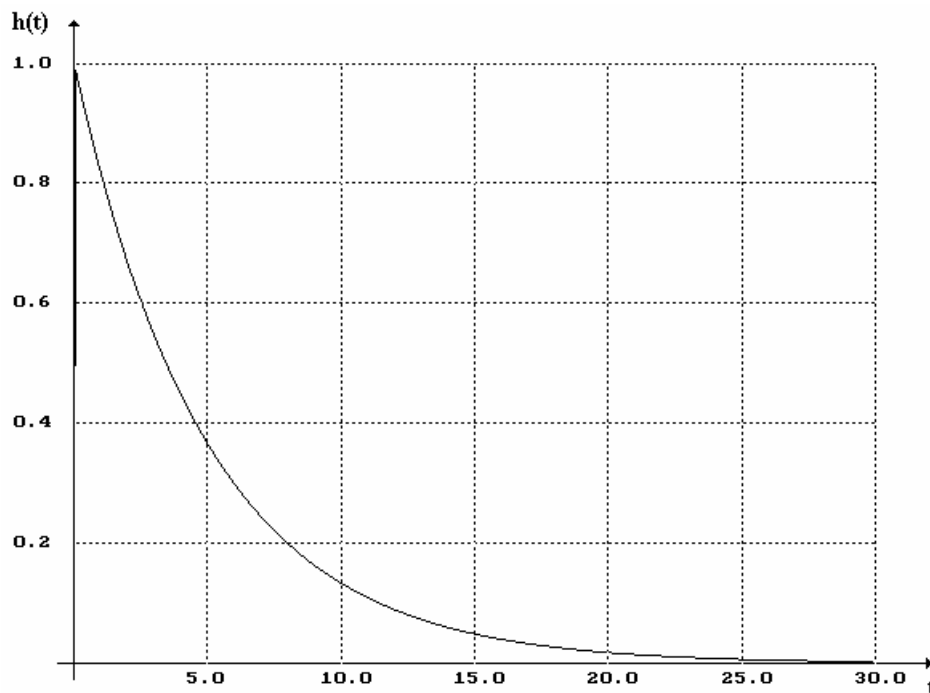


Рис. 1.6. Переходная характеристика реального дифференциального звена,
 $W(p) = 5p/(5p + 1)$.

Интегро-дифференциальное звено. Исходя из передаточной функции (1.67) и формулы (1.36) имеем

$$\bar{h}(p) = k \frac{\alpha T p + 1}{(T p + 1) p}. \quad (1.76)$$

Среди рассмотренных нами соответствий между оригиналами и изображениями - изображения типа полученного в последнем выражении (1.76) нет. Разложим правую часть (1.76) на элементарные слагаемые

$$k \frac{\alpha T p + 1}{(T p + 1) p} \equiv \frac{A_0}{p} + \frac{A_1}{T p + 1}. \quad (1.77)$$

Помножим левую и правую часть полученного тождества на знаменатель при искомом коэффициенте A_0 . После сокращения получим

$$k \frac{\alpha T p + 1}{T p + 1} \equiv A_0 + p \frac{A_1}{T p + 1}.$$

А теперь положим в этом тождестве $p = 0$ (тождество должно выполняться при любых значениях аргумента, почему бы ему не выполниться при $p = 0$?). Имеем

$$A_0 = k. \quad (1.78)$$

Имея значение A_0 из (1.77) можно получить A_1 . Но можно поступить так же, как при определении A_0 . Помножим (1.77) на $(Tp+1)$, то есть на знаменатель при искомом коэффициенте A_1

$$k \frac{\alpha Tp + 1}{p} \equiv \frac{A_0}{p} (Tp + 1) + A_1.$$

При $Tp + 1 = 0$ будем иметь

$$A_1 = k \frac{\alpha Tp + 1}{p} \Big|_{p = -\frac{1}{T}} = -kT(1 - \alpha).$$

Подставляем найденные значения A_0 и A_1 в (1.77)

$$k \frac{\alpha Tp + 1}{(Tp + 1)p} = \frac{k}{p} - \frac{kT(1 - \alpha)}{Tp + 1}$$

или

$$k \frac{\alpha Tp + 1}{(Tp + 1)p} = k \frac{1}{p} - k(1 - \alpha) \frac{1}{p + \frac{1}{T}}.$$

Заменяем в (1.76) правую часть ее значением из последнего выражения.

Учитывая соответствия (1.7) и (1.8) имеем

$$h(t) = k[1 - (1 - \alpha)e^{-\frac{t}{T}}]. \quad (1.79)$$

Здесь, как и в формуле (1.75) общий множитель $1(t)$ опущен (по умолчанию).

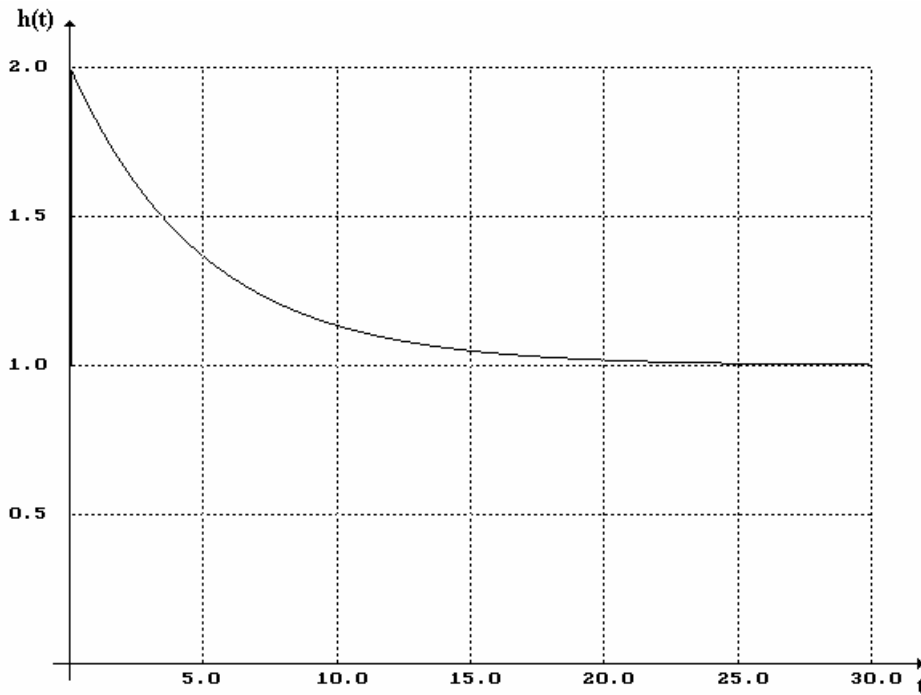


Рис. 1.7. Переходная характеристика интегро-дифференциального звена для $\alpha > 1$, $W(p) = (10p + 1)/(5p + 1)$

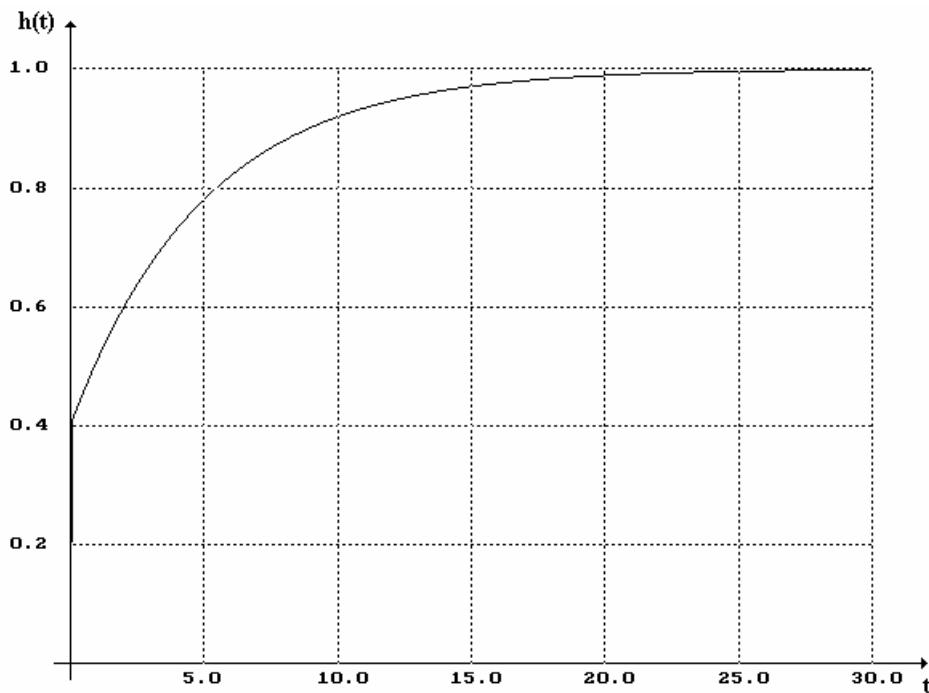


Рис. 1.8. Переходная характеристика интегро-дифференциального звена для $\alpha > 1$, $W(p) = (2p + 1)/(5p + 1)$

Апериодическое звено I порядка можно рассматривать как частный случай интегро-дифференциального звена при $\alpha = 0$, итак,

$$h(t) = k(1 - e^{-\frac{t}{T}}). \quad (1.80)$$

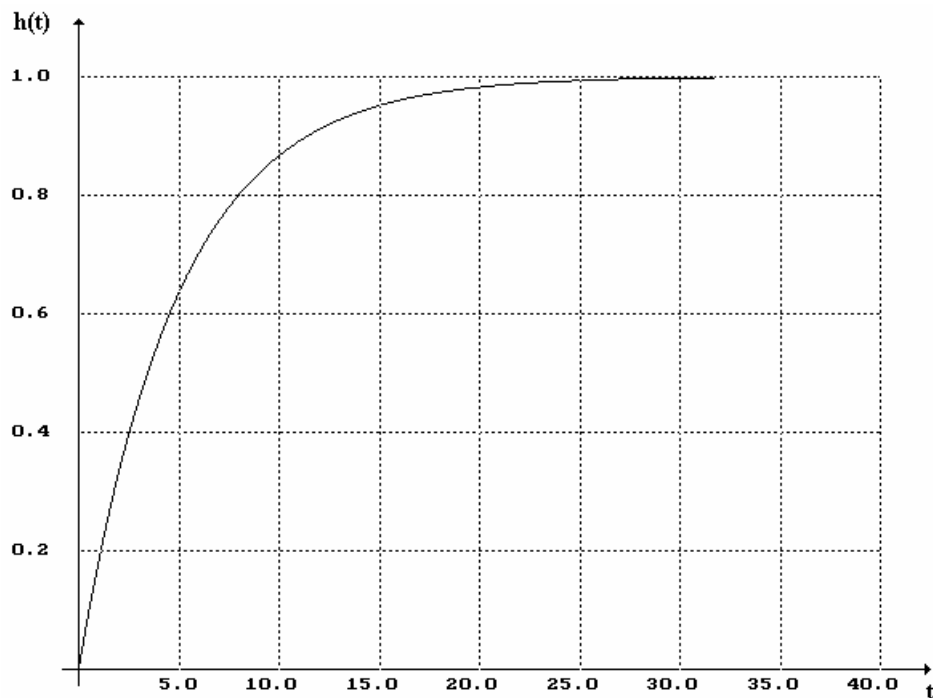


Рис. 1.9. Переходная характеристика аperiodического звена первого порядка, $W(p) = 1/(5p + 1)$.

Для **aperiodического звена II порядка** передаточная функция задается формулой (1.59). Итак, изображение переходной характеристики

$$\bar{h}(p) = \frac{k}{(a_2 p^2 + a_1 p + 1)p}. \quad (1.81)$$

Чтобы разложить правую часть (1.81) на элементарные слагаемые, надо сначала определить корни полинома в знаменателе

$$p_{1,2} = \frac{-a_1 \pm \sqrt{a_1^2 - 4a_2}}{2a_2}. \quad (1.82)$$

В определении аperiodического звена отмечается, про $a_1^2 - 4a_2 \geq 0$.

Рассмотрим сначала случай, когда $a_1^2 - 4a_2 > 0$. Тогда $p_1 \neq p_2$ и (1.81) можно переписать так

$$\bar{h}(p) = \frac{\frac{k}{a_2}}{(p - p_1)(p - p_2)p}. \quad (1.83)$$

А теперь уже можно разлагать (1.83) на элементарные слагаемые

$$\frac{\frac{k}{a_2}}{(p-p_1)(p-p_2)p} \equiv \frac{B_0}{p} + \frac{B_1}{p-p_1} + \frac{B_2}{p-p_2}. \quad (1.84)$$

Коэффициенты B_0 , B_1 , B_2 будем определять последовательно как было показано выше

$$B_0 = \left. \frac{\frac{k}{a_2}}{(p-p_1)(p-p_2)p} \right|_{p=0} = \left. \frac{k}{a_2 p^2 + a_1 p + 1} \right|_{p=0} = k,$$

$$B_1 = \left. \frac{\frac{k}{a_2}}{(p-p_2)p} \right|_{p=p_1} = \frac{\frac{k}{a_2}}{(p_1-p_2)p_1},$$

$$B_2 = \left. \frac{\frac{k}{a_2}}{(p-p_1)p} \right|_{p=p_2} = \frac{\frac{k}{a_2}}{(p_2-p_1)p_2}.$$

Подставляя найденные значения коэффициентов в (1.83), а потом в (1.82) и используя соответствия (1.7) и (1.8) будем иметь

$$h(t) = k \left[1 + \frac{1}{a_2} \frac{1}{p_1 - p_2} \left(\frac{e^{p_1 t}}{p_1} - \frac{e^{p_2 t}}{p_2} \right) \right]. \quad (1.85)$$

Раньше мы договорились считать все коэффициенты элементарных звеньев неотрицательными, тогда согласно (1.82) $p_1 < 0$ и $p_2 < 0$. При $t \rightarrow \infty$ экспоненты в (1.85) будут приближаться к нулю и $h(t)|_{t \rightarrow \infty} = k$.

Рассмотрим теперь случай, когда $a_1^2 - 4a_2 = 0$. Тогда согласно (1.82)

$$p_1 = p_2 = \frac{-a_1}{2a_2} \quad (1.86)$$

и изображение (1.81) разлагается на элементарные слагаемые таким образом

$$\frac{\frac{k}{a_2}}{(p-p_1)^2 p} \equiv \frac{C_0}{p} + \frac{C_1}{p-p_1} + \frac{C_2}{(p-p_1)^2}. \quad (1.87)$$

Очевидно, что,

$$C_2 = \left. \frac{k/a_2}{p} \right|_{p=p_1} = \frac{k}{a_2 p_1}, \quad C_0 = \left. \frac{k/a_2}{(p-p_1)^2} \right|_{p=0} = \frac{k}{a_2 p_1^2} = k.$$

Коэффициент C_1 , конечно, можно определить из (1.87), поскольку C_0 и C_2 уже известны. Но мы продемонстрируем другой подход. Если попробовать умножить левую и правую часть (1.87) на знаменатель при этом коэффициенте (общий подход), то получим

$$\frac{k}{(p-p_1)p} = \frac{C_0}{p}(p-p_1) + C_1 + \frac{C_2}{p-p_1}.$$

Попытка рассмотреть полученное выражение при $p-p_1=0$ (то есть при $p=p_1$) приводит к неопределенности типа $\infty = \infty$. Чтобы не усложнять себе жизнь раскрытием неопределенностей, поступим иначе. Перенесем в (1.84) слагаемое с уже определенным C_2 в левую часть

$$\frac{k}{(p-p_1)^2 p} - \frac{k}{p_1(p-p_1)^2} = \frac{C_0}{p} + \frac{C_1}{p-p_1}.$$

Приведем левую часть к общему знаменателю

$$\frac{k}{a_2} \cdot \frac{p_1-p}{(p-p_1)^2 p \cdot p_1} = \frac{C_0}{p} + \frac{C_1}{p-p_1}.$$

После очевидного сокращения получим

$$-\frac{k}{a_2(p-p_1)p \cdot p_1} = \frac{C_0}{p} + \frac{C_1}{p-p_1}.$$

Из последнего выражения C_1 находится по общей схеме

$$C_1 = - \left. \frac{k}{a_2 p \cdot p_1} \right|_{p=p_1} = - \frac{k}{a_2 p_1^2}.$$

Подставляем в (1.84) найденные значения коэффициентов, после чего (1.81) приобретает вид

$$\bar{h}(p) = k \left\{ \frac{1}{p} - \frac{1}{a_2 p_1^2} \left[\frac{1}{p - p_1} - \frac{p_1}{(p - p_1)^2} \right] \right\}.$$

Если теперь кроме прежде использовавшихся соответствий (1.7) и (1.8) использовать еще и (1.12) при $k = 1$, то из последнего выражения следует

$$h(t) = k \left[1 - \frac{1}{a_2 p_1^2} (1 - p_1 t) e^{-p_1 t} \right]. \quad (1.88)$$

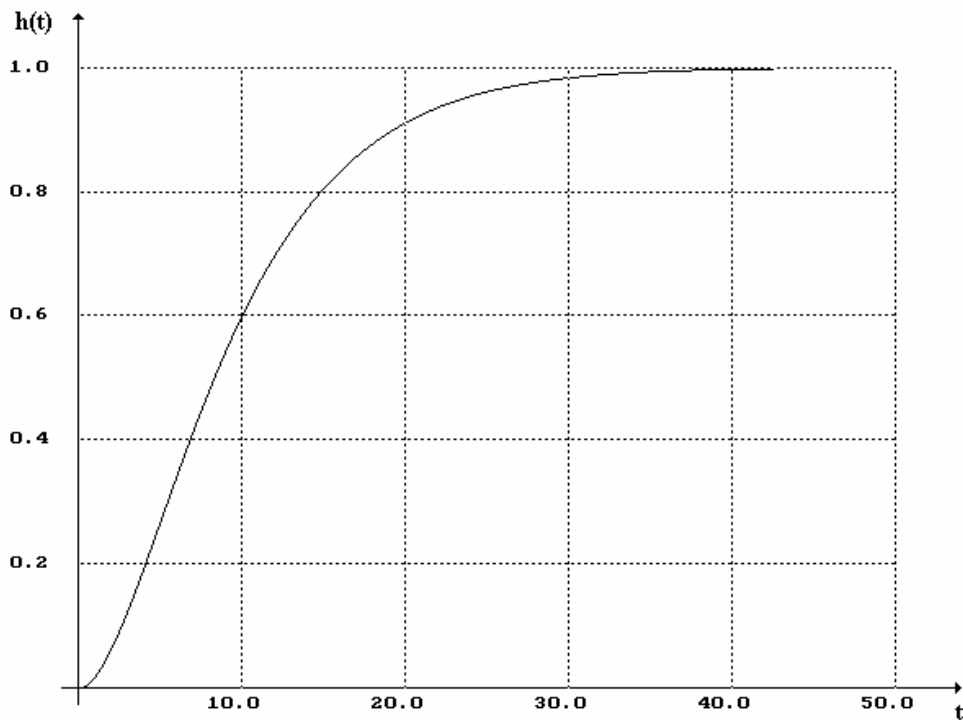


Рис. 1.10. Переходная характеристика аperiodического звена второго порядка,

$$W(p) = \frac{1}{25p^2 + 10p + 1}.$$

Для **колебательного звена** $a_1^2 - 4a_2 < 0$, корни полинома $a_2 p^2 + a_1 p + 1$ будут согласно (1.82)

$$p_{1,2} = \alpha \pm j\beta, \quad (1.89)$$

где $\alpha = \frac{-a_1}{2a_2}$, $\beta = \frac{\sqrt{4a_2 - a_1^2}}{2a_2}$, $j = \sqrt{-1}$.

Изображение $\bar{h}(p)$ будет

$$\bar{h}(p) = \frac{\frac{k}{a_2}}{(p - \alpha - j\beta)(p - \alpha + j\beta)p}. \quad (1.90)$$

Разложим его по обыкновению на элементарные слагаемые

$$\frac{\frac{k}{a_2}}{(p - \alpha - j\beta)(p - \alpha + j\beta)p} = \frac{D_0}{p} + \frac{D_1}{p - \alpha - j\beta} + \frac{D_2}{p - \alpha + j\beta}. \quad (1.91)$$

Определяем коэффициенты

$$D_0 = \left. \frac{\frac{k}{a_2}}{(p - \alpha - j\beta)(p - \alpha + j\beta)} \right|_{p=0} = \left. \frac{k}{a_2 p^2 + a_1 p + 1} \right|_{p=0} = k,$$

$$D_1 = \left. \frac{\frac{k}{a_2}}{(p - \alpha + j\beta)p} \right|_{p=\alpha+j\beta} = \frac{\frac{k}{a_2}}{2j\beta(\alpha + j\beta)} = \frac{k}{2a_2\beta(-\beta + j\alpha)} = \frac{-k}{2a_2\beta(\beta - j\alpha)} = \frac{-k(\beta + j\alpha)}{2a_2\beta(\beta^2 + \alpha^2)}.$$

Вычислим значение $\beta^2 + \alpha^2$

$$\beta^2 + \alpha^2 = \frac{4a_2 - a_1^2}{4a_2^2} + \frac{a_1^2}{4a_2^2} = \frac{1}{a_2}.$$

Итак,

$$D_1 = \frac{-k(\beta + j\alpha)}{2\beta} = -\frac{k}{2} \left(1 + j \frac{\alpha}{\beta} \right).$$

D_2 определяется аналогичным образом. Опустим вычисления

$$D_2 = -\frac{k}{2} \left(1 - j \frac{\alpha}{\beta} \right).$$

Подставим теперь значения коэффициентов в (1.90), а потом в (1.89)

$$\bar{h}(p) = k \left[\frac{1}{p} - \frac{1}{2} \left(\frac{1 + j \frac{\alpha}{\beta}}{p - \alpha - j\beta} + \frac{1 - j \frac{\alpha}{\beta}}{p - \alpha + j\beta} \right) \right].$$

Переходим к оригиналам согласно формул a_m (1.7), (1.8), рассматривая комплексные корни как реализации коэффициента a в (1.8)

$$h(t) = k \left\{ 1 - \frac{1}{2} \left[\left(1 + j \frac{\alpha}{\beta} \right) e^{(\alpha + j\beta)t} + \left(1 - j \frac{\alpha}{\beta} \right) e^{(\alpha - j\beta)t} \right] \right\}.$$

Упростим выражение в квадратных скобках

$$\begin{aligned} & \left(1 + j \frac{\alpha}{\beta} \right) e^{(\alpha + j\beta)t} + \left(1 - j \frac{\alpha}{\beta} \right) e^{(\alpha - j\beta)t} = \\ & = e^{\alpha t} \left[\left(1 + j \frac{\alpha}{\beta} \right) (\cos \beta t + j \sin \beta t) + \left(1 - j \frac{\alpha}{\beta} \right) (\cos \beta t - j \sin \beta t) \right] = \\ & = e^{\alpha t} \left[\cos \beta t + j \sin \beta t + j \frac{\alpha}{\beta} \cos \beta t - \frac{\alpha}{\beta} \sin \beta t + \cos \beta t - j \sin \beta t - j \frac{\alpha}{\beta} \cos \beta t - \frac{\alpha}{\beta} \sin \beta t \right]. \end{aligned}$$

После сокращения и сведения подобных имеем

$$\left(1 + j \frac{\alpha}{\beta} \right) e^{(\alpha + j\beta)t} + \left(1 - j \frac{\alpha}{\beta} \right) e^{(\alpha - j\beta)t} = 2e^{\alpha t} \left(\cos \beta t - \frac{\alpha}{\beta} \sin \beta t \right).$$

Подставляем результат в (1.91)

$$h(t) = k \left[1 - e^{\alpha t} \left(\cos \beta t - \frac{\alpha}{\beta} \sin \beta t \right) \right]. \quad (1.92)$$

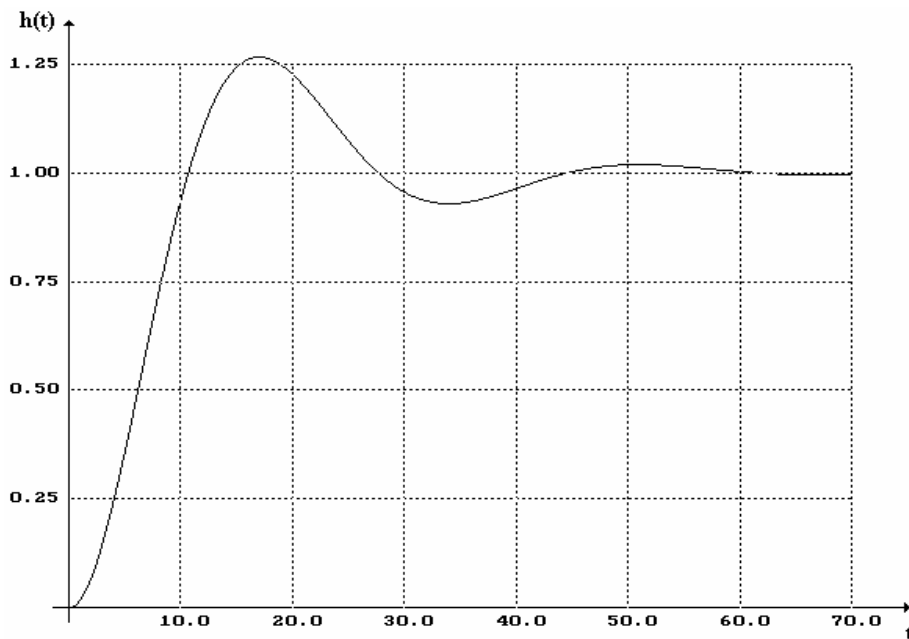


Рис. 1.11. Переходная характеристика колебательного звена,

$$W(p) = \frac{1}{25p^2 + 5p + 1}.$$

Консервативное звено – это частный случай колебательного звена, когда $a_1 = 0$. Итак, согласно (1.89)

$$p_{1,2} = \pm j\beta, \quad (1.93)$$

поскольку $\alpha = 0$, $\beta = \frac{1}{\sqrt{a_2}}$.

Тогда из (1.92) получим

$$h(t) = k(1 - \cos\beta t). \quad (1.94)$$

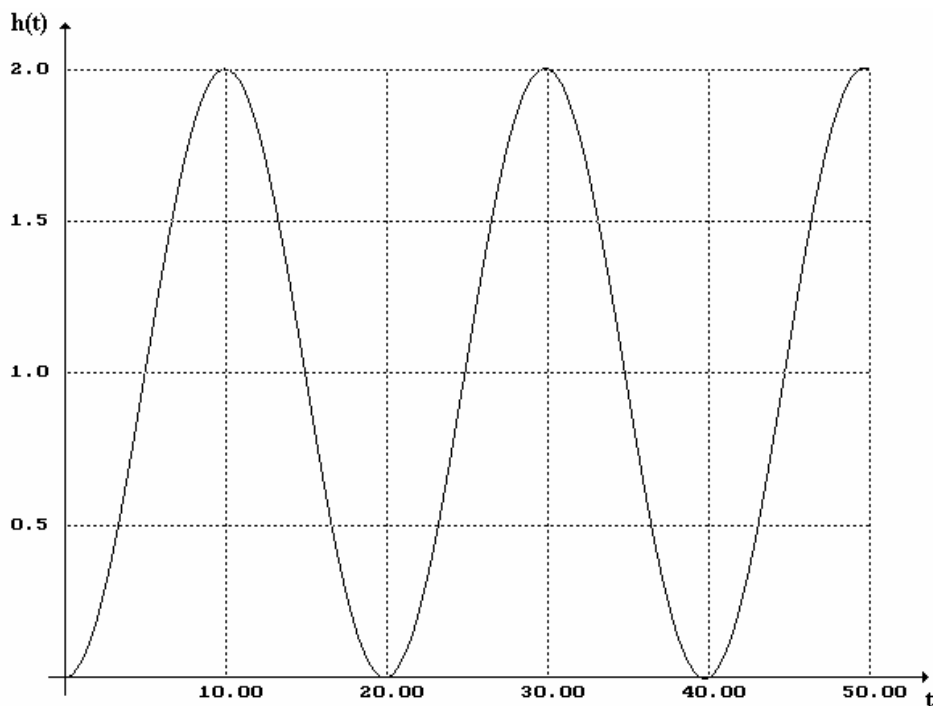


Рис. 1.12. Переходная характеристика консервативного звена, $W(p) = \frac{1}{10p^2 + 1}$.

Звено транспортного запаздывания в соответствии с определением просто задерживает входной сигнал на время τ , ведь

$$h(t) = 1(t - \tau). \quad (1.95)$$

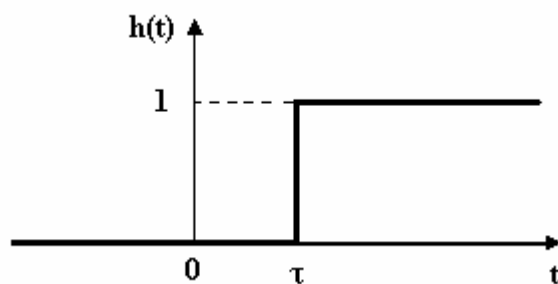


Рис. 1.13. Переходная характеристика звена транспортного запаздывания

1.7. Частотные характеристики элементарных динамических звеньев

Используя формулы (1.44)-(1.52) для элементарных звеньев будем иметь:

Безынерционное звено

$$\begin{cases} W(p) = k; & W(j\omega) = k; \\ A(\omega) = k; & \varphi(\omega) = 0; \\ \operatorname{Re}(\omega) = k; & \operatorname{Im}(\omega) = 0. \end{cases} \quad (1.96)$$

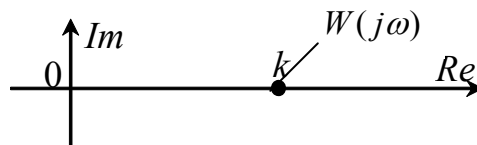


Рис. 1.14. Годограф амплитудно-фазовой характеристики безынерционного звена

Интегральное звено

$$\begin{cases} W(p) = \frac{k}{p}; & W(j\omega) = \frac{k}{j\omega} = -j\frac{k}{\omega}; \\ A(\omega) = \frac{k}{\omega}; & \varphi(\omega) = -\frac{\pi}{2}; \\ \operatorname{Re}(\omega) = 0; & \operatorname{Im}(\omega) = -\frac{k}{\omega}. \end{cases} \quad (1.97)$$

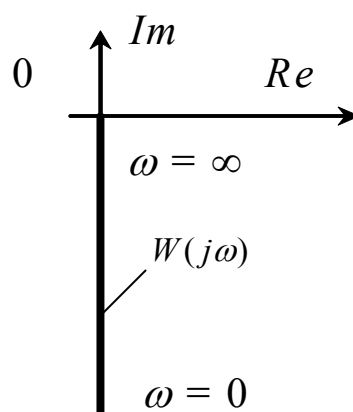


Рис. 1.15. Годограф амплитудно-фазовой характеристики интегрального звена

Идеальное дифференциальное звено

$$\begin{cases} W(p) = kp; & W(j\omega) = jk\omega; \\ A(\omega) = k\omega; & \varphi(\omega) = \frac{\pi}{2}; \\ \operatorname{Re}(\omega) = 0; & \operatorname{Im}(\omega) = k\omega. \end{cases} \quad (1.98)$$

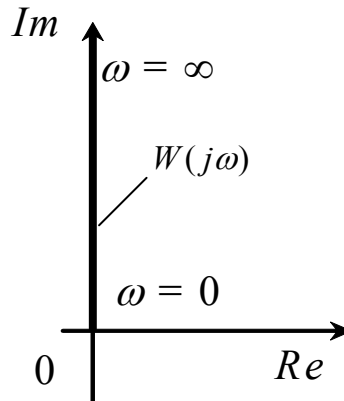


Рис. 1.16. Годограф амплитудно-фазовой характеристики идеального дифференциального звена.

Реальное дифференциальное звено

$$\begin{cases} W(p) = k \frac{Tp}{Tp+1}; & W(j\omega) = k \frac{jT\omega}{1+jT\omega} = k \frac{(T\omega)^2 + jT\omega}{1+(T\omega)^2}; \\ A(\omega) = k \frac{T\omega}{\sqrt{1+(T\omega)^2}}; & \varphi(\omega) = \frac{\pi}{2} - \operatorname{arctg}(T\omega); \\ \operatorname{Re}(\omega) = k \frac{(T\omega)^2}{1+(T\omega)^2}; & \operatorname{Im}(\omega) = k \frac{T\omega}{1+(T\omega)^2}. \end{cases} \quad (1.99)$$

Годографы амплитудно-фазовых характеристик трех предыдущих звеньев были очевидными (точка, участок мнимой оси).

Для данного звена попробуем выявить форму годографа. Для этого поделим $\operatorname{Re}(\omega)$ на $\operatorname{Im}(\omega)$. Получим:

$$\frac{\operatorname{Re}(\omega)}{\operatorname{Im}(\omega)} = T\omega.$$

А теперь полученное значение $T\omega$ подставим, например, в формулу для $\operatorname{Re}(\omega)$:

$$\operatorname{Re}(\omega) = k \frac{\frac{\operatorname{Re}^2(\omega)}{\operatorname{Im}^2(\omega)}}{1 + \frac{\operatorname{Re}^2(\omega)}{\operatorname{Im}^2(\omega)}}.$$

Левую и правую части полученного выражения делим на $\operatorname{Re}(\omega)$ и упрощаем

$$1 = k \frac{\operatorname{Re}(\omega)}{\operatorname{Re}^2(\omega) + \operatorname{Im}^2(\omega)},$$

или, что то же

$$\operatorname{Re}^2(\omega) + \operatorname{Im}^2(\omega) = k \operatorname{Re}(\omega).$$

Переносим $k \operatorname{Re}(\omega)$ в левую часть и дополняем до полного квадрата:

$$\operatorname{Re}^2(\omega) - 2 \frac{k}{2} \operatorname{Re}(\omega) + \left(\frac{k}{2}\right)^2 + \operatorname{Im}^2(\omega) = \left(\frac{k}{2}\right)^2.$$

Окончательно имеем:

$$\left(\operatorname{Re}(\omega) - \frac{k}{2}\right)^2 + \operatorname{Im}^2(\omega) = \left(\frac{k}{2}\right)^2 \quad (1.100)$$

Получено уравнение окружности радиусом $\frac{k}{2}$ с центром на вещественной оси

на расстоянии $\frac{k}{2}$ от начала координат (справа).

Выполняем элементарный анализ:

$$\operatorname{Re}(\omega)|_{\omega=0} = k \frac{(T\omega)^2}{1+(T\omega)^2} \Big|_{\omega=0} = 0,$$

$$\operatorname{Im}(\omega)|_{\omega=0} = k \frac{T\omega}{1+(T\omega)^2} \Big|_{\omega=0} = 0,$$

$$\operatorname{Re}(\omega)|_{\omega=\infty} = k \frac{(T\omega)^2}{1+(T\omega)^2} \Big|_{\omega=\infty} = k,$$

$$\operatorname{Im}(\omega)|_{\omega=\infty} = k \frac{T\omega}{1+(T\omega)^2} \Big|_{\omega=\infty} = 0,$$

$$\operatorname{Re}(\omega)|_{T\omega=1} = k \frac{(T\omega)^2}{1+(T\omega)^2} \Big|_{T\omega=1} = \frac{k}{2},$$

$$\operatorname{Im}(\omega)|_{T\omega=1} = k \frac{T\omega}{1+(T\omega)^2} \Big|_{T\omega=1} = \frac{k}{2},$$

$$\operatorname{Im}(\omega)|_{\omega>0} > 0.$$

Итак, в диапазоне частот $0 < \omega < \infty$ это полуокружность в I квадранте, годограф начинается ($\omega = 0$) в начала координат, заканчивается ($\omega = \infty$) в точке

$k + j0$, его верхняя точка отвечает частоте $\omega = \frac{1}{T}$.

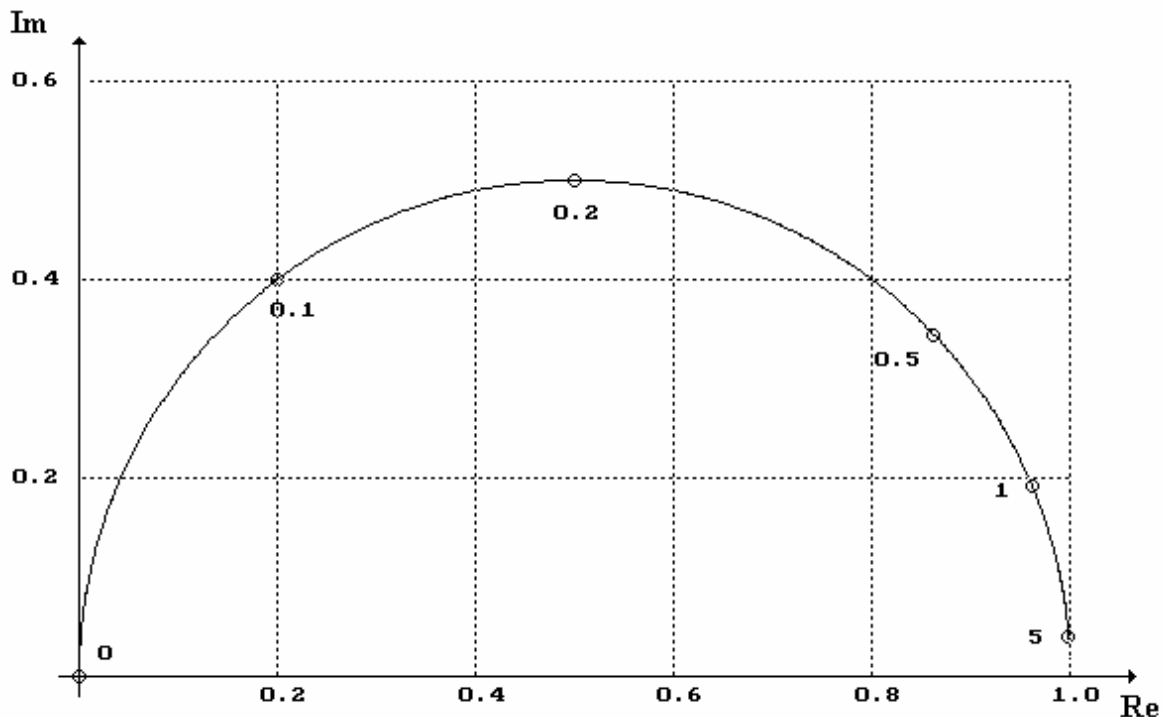


Рис. 1.17. Годограф амплитудно-фазовой характеристики реального дифференциального звена $W(p) = 5p/(5p+1)$.

Интегро-дифференциальное звено

$$\left\{ \begin{array}{l} W(p) = k \frac{\alpha T p + 1}{T p + 1}; \\ W(j\omega) = k \frac{1 + j\alpha T\omega}{1 + jT\omega} = k \frac{1 + \alpha(T\omega)^2 + j(\alpha - 1)T\omega}{1 + (T\omega)^2}; \\ A(\omega) = k \sqrt{\frac{1 + (\alpha T\omega)^2}{1 + (T\omega)^2}}; \\ \varphi(\omega) = \operatorname{arctg}(\alpha T\omega) - \operatorname{arctg}(T\omega); \\ \operatorname{Re}(\omega) = k \frac{1 + \alpha(T\omega)^2}{1 + (T\omega)^2}; \\ \operatorname{Im}(\omega) = k(\alpha - 1) \frac{T\omega}{1 + (T\omega)^2}. \end{array} \right. \quad (1.101)$$

Выявление формы годографа $W(j\omega)$ начнем с определения $\operatorname{Re}(\omega) - k$:

$$\operatorname{Re}(\omega) - k = k \frac{1 + \alpha(T\omega)^2}{1 + (T\omega)^2} - k = k(\alpha - 1) \frac{(T\omega)^2}{1 + (T\omega)^2}.$$

А теперь поделим $\operatorname{Re}(\omega) - k$ на $\operatorname{Im}(\omega)$:

$$\frac{\operatorname{Re}(\omega) - k}{\operatorname{Im}(\omega)} = T\omega.$$

Полученное значение $T\omega$ подставим в формулу для $\operatorname{Im}(\omega)$.

$$\operatorname{Im}(\omega) = k(\alpha - 1) \frac{\frac{\operatorname{Re}(\omega) - k}{\operatorname{Im}(\omega)}}{1 + \frac{[\operatorname{Re}(\omega) - k]^2}{\operatorname{Im}^2(\omega)}}.$$

Упрощаем выражение в правой части:

$$\operatorname{Im}(\omega) = k(\alpha - 1) \frac{[\operatorname{Re}(\omega) - k]\operatorname{Im}(\omega)}{[\operatorname{Re}(\omega) - k]^2 + \operatorname{Im}^2(\omega)}.$$

Делим левую и правую части на $\operatorname{Im}(\omega)$:

$$1 = k(\alpha - 1) \frac{\operatorname{Re}(\omega) - k}{[\operatorname{Re}(\omega) - k]^2 + \operatorname{Im}^2(\omega)}$$

откуда

$$[\operatorname{Re}(\omega) - k]^2 + \operatorname{Im}^2(\omega) = k(\alpha - 1)[\operatorname{Re}(\omega) - k].$$

Выполняем очевидные преобразования:

$$[\operatorname{Re}(\omega) - k]^2 - 2\frac{k}{2}(\alpha - 1)[\operatorname{Re}(\omega) - k] + \left[\frac{k}{2}(\alpha - 1)\right]^2 + \operatorname{Im}^2(\omega) = \left[\frac{k}{2}(\alpha - 1)\right]^2$$

После упрощения имеем:

$$\left[\operatorname{Re}(\omega) - \frac{k}{2}(\alpha + 1)\right]^2 + \operatorname{Im}^2(\omega) = \left[\frac{k}{2}(\alpha - 1)\right]^2. \quad (1.102)$$

Это опять же уравнение окружности с центром на вещественной оси в точке

$$\operatorname{Re}(\omega) = \frac{k}{2}(\alpha + 1) \text{ и радиусом } R = \left|\frac{k}{2}(\alpha - 1)\right|.$$

Выполним элементарный анализ:

$$\operatorname{Re}(\omega)\Big|_{\omega=0} = k \frac{1 + \alpha(T\omega)^2}{1 + (T\omega)^2}\Big|_{\omega=0} = k,$$

$$\operatorname{Im}(\omega)\Big|_{\omega=0} = k(\alpha - 1) \frac{T\omega}{1 + (T\omega)^2}\Big|_{\omega=0} = 0,$$

$$\operatorname{Re}(\omega)\Big|_{\omega=\infty} = k \frac{1 + \alpha(T\omega)^2}{1 + (T\omega)^2}\Big|_{\omega=\infty} = k\alpha,$$

$$\operatorname{Im}(\omega)\Big|_{\omega=\infty} = k(\alpha - 1) \frac{T\omega}{1 + (T\omega)^2}\Big|_{\omega=\infty} = 0.$$

Пусть, например, $\alpha > 1$, тогда $R = \frac{k}{2}(\alpha - 1)$ и в диапазоне $0 < \omega \leq \infty$ $\operatorname{Im}(\omega) > 0$,

итак, имеем полуокружность в I квадранте, тогда как при $\alpha < 1$ радиус будет

$R = \frac{k}{2}(1 - \alpha)$, и в диапазоне $0 < \omega \leq \infty$ $\operatorname{Im}(\omega) < 0$, то есть окружность будет в IV

квадранте.

$$\operatorname{Re}(\omega)\Big|_{T\omega=1} = k \frac{1 + \alpha(T\omega)^2}{1 + (T\omega)^2}\Big|_{T\omega=1} = \frac{k}{2}(1 + \alpha),$$

И, в конце концов:

$$\operatorname{Im}(\omega)\Big|_{T\omega=1} = k(\alpha - 1) \frac{T\omega}{1 + (T\omega)^2}\Big|_{T\omega=1} = \frac{k}{2}(\alpha - 1),$$

что значит, что наивысшая (при $\alpha > 1$) или, соответственно, наинизшая (при

$\alpha < 1$) точки годографа имеют место при частоте $\omega = \frac{1}{T}$.

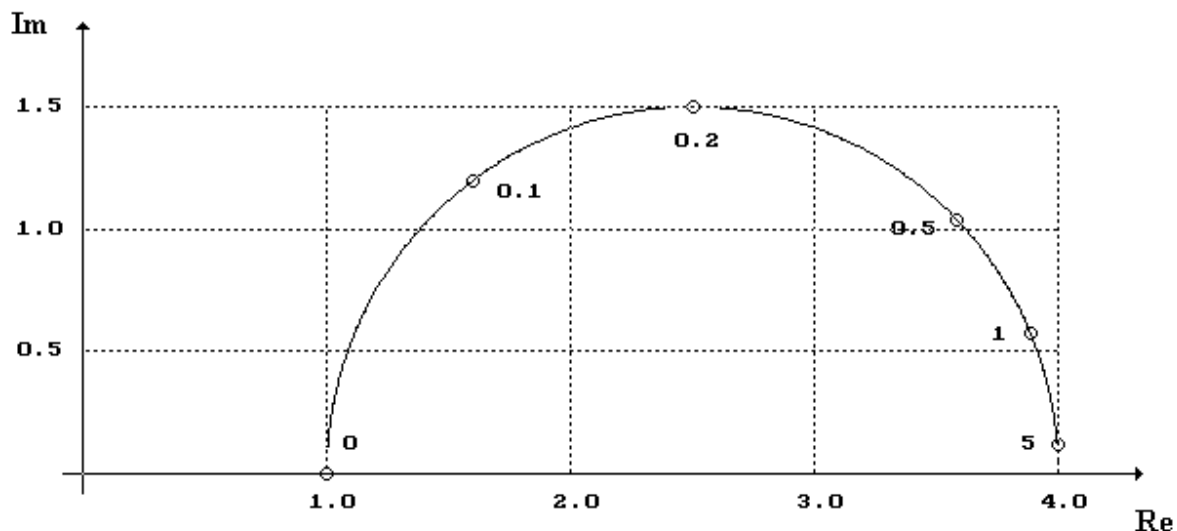


Рис. 1.18. Годограф амплитудно-фазовой характеристики интегро-дифференциального звена для $\alpha > 1$, $W(p) = (20p + 1)/(5p + 1)$.

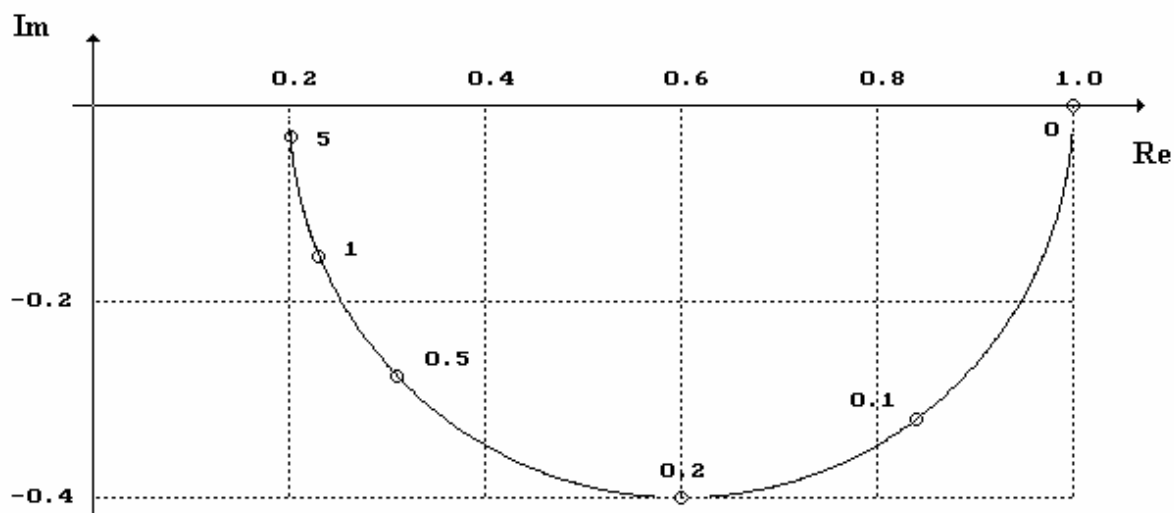


Рис. 1.19. Годограф амплитудно-фазовой характеристики интегро-дифференциального звена для $\alpha < 1$, $W(p) = (p + 1)/(5p + 1)$.

Апериодическое звено I порядка

$$\left\{ \begin{array}{l} W(p) = \frac{k}{Tp + 1}; \\ A(\omega) = \frac{k}{\sqrt{1 + (T\omega)^2}}; \\ \text{Re}(\omega) = \frac{k}{1 + (T\omega)^2}; \end{array} \right. \quad \left\{ \begin{array}{l} W(j\omega) = \frac{k}{1 + jT\omega} = k \frac{1 - jT\omega}{1 + (T\omega)^2}; \\ \varphi(\omega) = -\text{arctg}(T\omega); \\ \text{Im}(\omega) = -k \frac{T\omega}{1 + (T\omega)^2}. \end{array} \right. \quad (1.103)$$

А теперь вспомним, что аperiодическое звено I порядка может быть

получено из интегро-дифференциального при $\alpha = 0$ (это вариант, когда $\alpha < 1$), тогда уравнение (1.102) приобретает вид:

$$\left[\operatorname{Re}(\omega) - \frac{k}{2} \right]^2 + \operatorname{Im}^2(\omega) = \left(\frac{k}{2} \right)^2. \quad (1.104)$$

Оно ничем не отличается от прежде полученного уравнения (1.100) для реального дифференциального звена. Но различие однако есть. В диапазоне частот $0 < \omega \leq \infty$ $\operatorname{Im}(\omega) < 0$, итак при положительных частотах – это полуокружность в IV квадранте.

$$\begin{aligned} \operatorname{Re}(\omega)|_{\omega=0} &= k; & \operatorname{Im}(\omega)|_{\omega=0} &= 0; \\ \operatorname{Re}(\omega)|_{\omega=\infty} &= 0; & \operatorname{Im}(\omega)|_{\omega=\infty} &= 0; \\ \operatorname{Re}(\omega)|_{T\omega=1} &= \frac{k}{2}; & \operatorname{Im}(\omega)|_{T\omega=1} &= -\frac{k}{2}. \end{aligned}$$

Итак, наиболее низкая точка годографа имеет место при $\omega = \frac{1}{T}$.

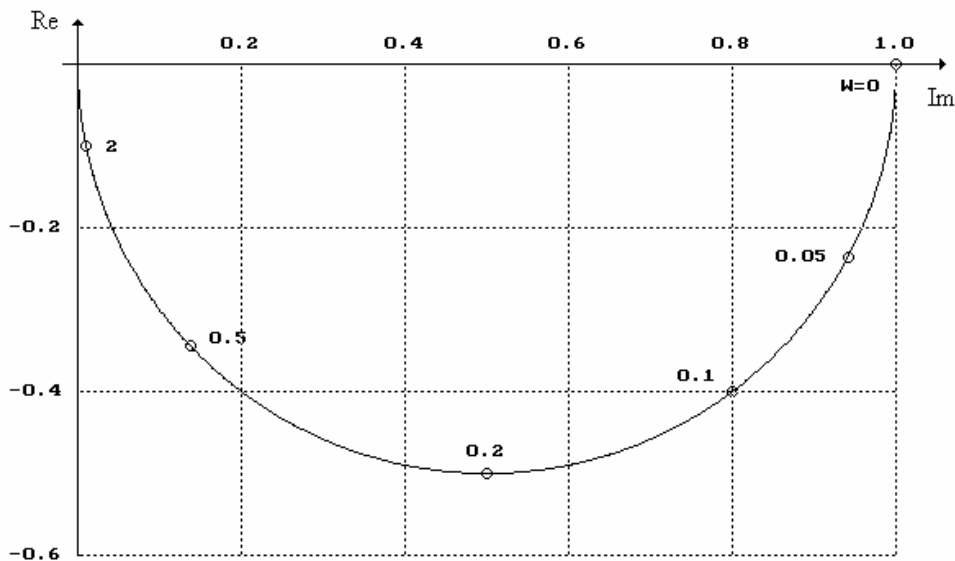


Рис.1.20. Годограф амплитудно-фазовой характеристики апериодического звена первого порядка, $W(p) = 1/(5p + 1)$.

Апериодическое звено II порядка и колебательное звено

Поскольку передаточные функции этих двух звеньев одинаковые, то будем рассматривать их частотные характеристики совместно.

$$\left\{ \begin{array}{l} W(p) = \frac{k}{a_2 p^2 + a_1 p + 1}; \\ W(j\omega) = \frac{k}{1 - a_2 \omega^2 + ja_1 \omega} = k \frac{(1 - a_2 \omega^2) - j(a_1 \omega)}{(1 - a_2 \omega^2)^2 + (a_1 \omega)^2}; \\ A(\omega) = \frac{k}{\sqrt{(1 - a_2 \omega^2)^2 + (a_1 \omega)^2}}; \\ \varphi(\omega) = -\operatorname{arctg} \frac{a_1 \omega}{1 - a_2 \omega^2}; \\ \operatorname{Re}(\omega) = k \frac{1 - a_2 \omega^2}{(1 - a_2 \omega^2)^2 + (a_1 \omega)^2}; \\ \operatorname{Im}(\omega) = -k \frac{a_1 \omega}{(1 - a_2 \omega^2)^2 + (a_1 \omega)^2}. \end{array} \right. \quad (1.105)$$

Анализ:

$$\begin{array}{ll} \operatorname{Re}(\omega)|_{\omega=0} = k; & \operatorname{Im}(\omega)|_{\omega=0} = 0; \\ \operatorname{Re}(\omega)|_{\omega=\infty} = 0; & \operatorname{Im}(\omega)|_{\omega=\infty} = 0; \\ \operatorname{Re}(\omega)|_{\omega=\frac{1}{\sqrt{a_2}}} = 0; & \operatorname{Im}(\omega)|_{\omega=\frac{1}{\sqrt{a_2}}} = -k \frac{\sqrt{a_2}}{a_1}; \\ \operatorname{Re}(\omega)|_{0 \leq \omega < \frac{1}{\sqrt{a_2}}} > 0; & \operatorname{Re}(\omega)|_{\omega > \frac{1}{\sqrt{a_2}}} < 0; \\ \operatorname{Im}(\omega)|_{0 < \omega \leq \infty} < 0. \end{array}$$

Итак, годограф находится в IV квадранте (где начинается) и в III квадранте (заканчивается).

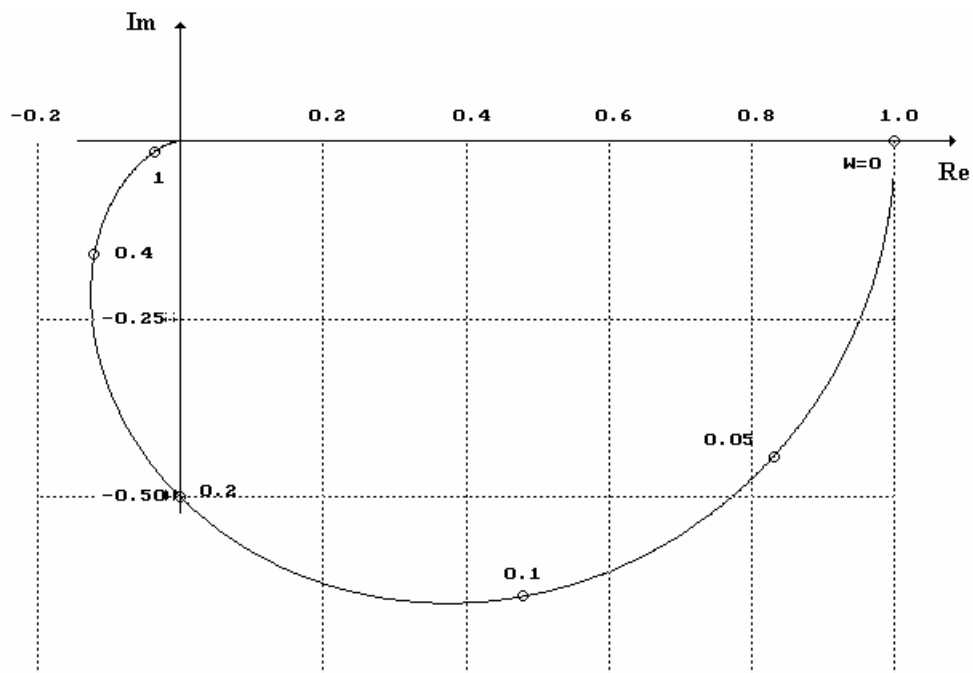


Рис.1.21. Годограф амплитудно-фазовой характеристики аperiodического звена второго порядка, $W(p) = 1/(25p^2 + 10p + 1)$

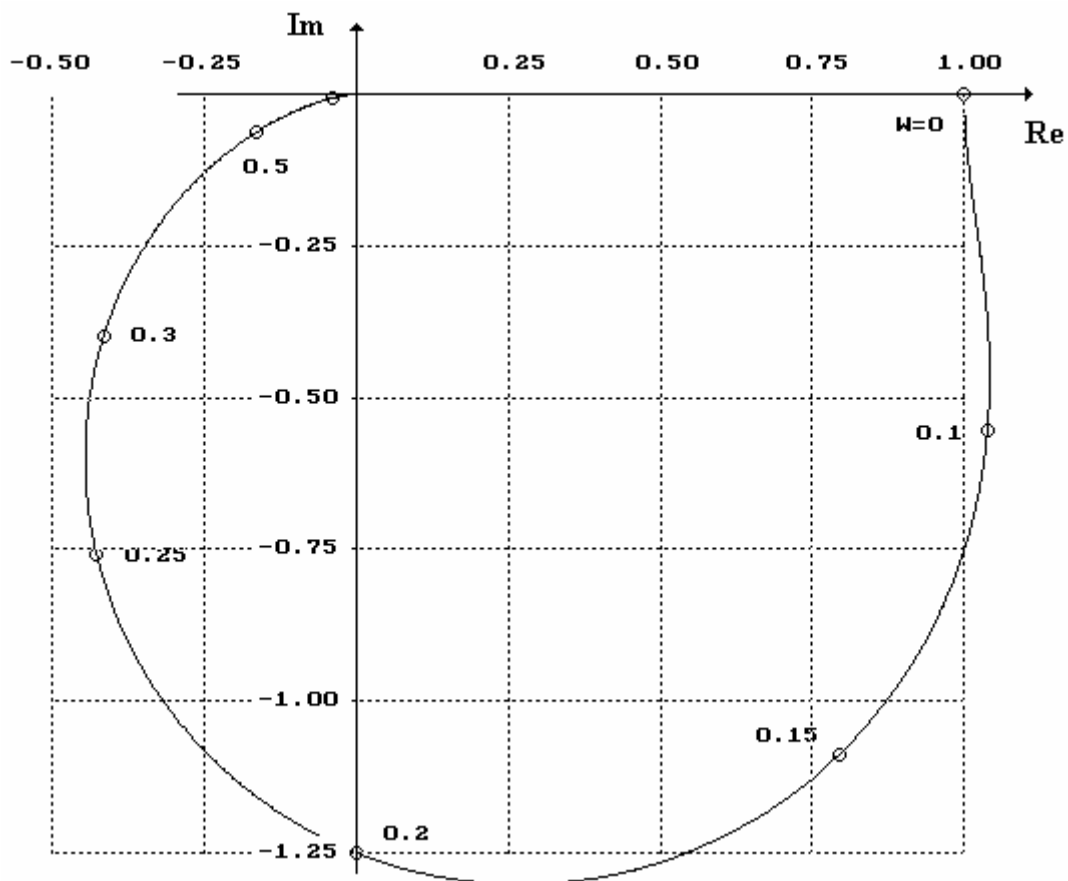


Рис.1.22. Годограф амплитудно-фазовой характеристики колебательного звена, $W(p) = 1/(25p^2 + 4p + 1)$

Консервативное звено

$$\left\{ \begin{array}{l} W(p) = \frac{k}{a_2 p^2 + 1}; \\ W(j\omega) = \frac{k}{1 - a_2 \omega^2}; \\ A(\omega) = \frac{k}{|1 - a_2 \omega^2|}; \\ \operatorname{Re}(\omega) = \frac{k}{1 - a_2 \omega^2}; \\ \operatorname{Im}(\omega) = 0; \\ \varphi(\omega) = \begin{cases} 0, & \omega < 1/\sqrt{a_2} \\ \pm \pi, & \omega > 1/\sqrt{a_2} \end{cases} \end{array} \right. \quad (1.106)$$

Итак, годограф $W(j\omega)$ консервативного звена лежит на вещественной оси.

Начинается в точке k и заканчивается в начале координат, поскольку:

$$\begin{aligned} \operatorname{Re}(\omega)|_{\omega=0} &= k; & \operatorname{Im}(\omega) &= 0; \\ \operatorname{Re}(\omega)|_{\omega=\infty} &= 0; & \operatorname{Im}(\omega) &= 0. \end{aligned}$$

Кроме того:

$$\begin{aligned} \operatorname{Re}(\omega) \Big|_{\omega > \frac{1}{\sqrt{a_2}}} &< 0; \\ \operatorname{Re}(\omega) \Big|_{\omega < \frac{1}{\sqrt{a_2}}} &> 0; \\ \operatorname{Re}(\omega) \Big|_{\omega = \frac{1}{\sqrt{a_2}}} &= \infty. \end{aligned}$$

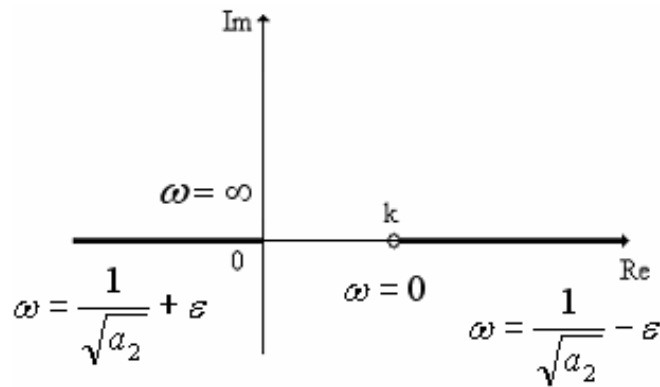


Рис.1.23. Годограф амплитудно-фазовой характеристики консервативного звена, ε - бесконечно малая величина

Представляет интерес сравнения амплитудно-частотных характеристик звеньев II порядка: апериодического, колебательного и консервативного.

Пусть $k = 1$; $a_2 = 4$; $a_1 = 5$ (апериодическое звено), $a_1 = 2$ (колебательное) и $a_1 = 0$ (консервативное).

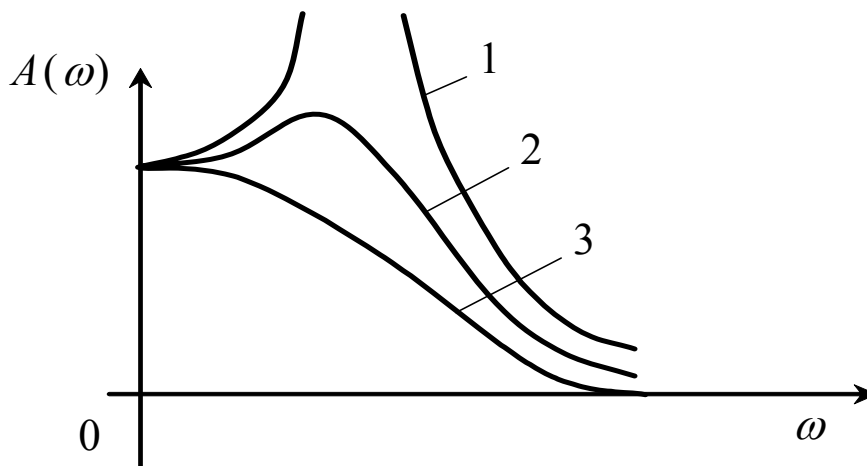


Рис. 1.24. Графики амплитудно-частотных характеристик: 1-консервативного, 2-колебательного и 3-апериодического звена II-го порядка.

Звено транспортного запаздывания

$$\left\{ \begin{array}{l} W(p) = e^{-p\tau}; \\ W(j\omega) = e^{-j\omega\tau} = \cos(\omega\tau) - j \sin(\omega\tau); \\ A(\omega) = 1; \\ \varphi(\omega) = -\omega\tau; \\ \operatorname{Re}(W) = \cos(\omega\tau); \\ \operatorname{Im}(W) = -\sin(\omega\tau). \end{array} \right. \quad (1.107)$$

Это, очевидно, окружность единичного радиуса с центром в начале координат комплексной плоскости. Начинается она в точке $\text{Re}(\omega)|_{\omega=0} = k$; $\text{Im}(\omega)|_{\omega=0} = 0$. Заканчивается (при $\omega = \infty$) где-то на окружности, а вот где именно – смотрите сами.

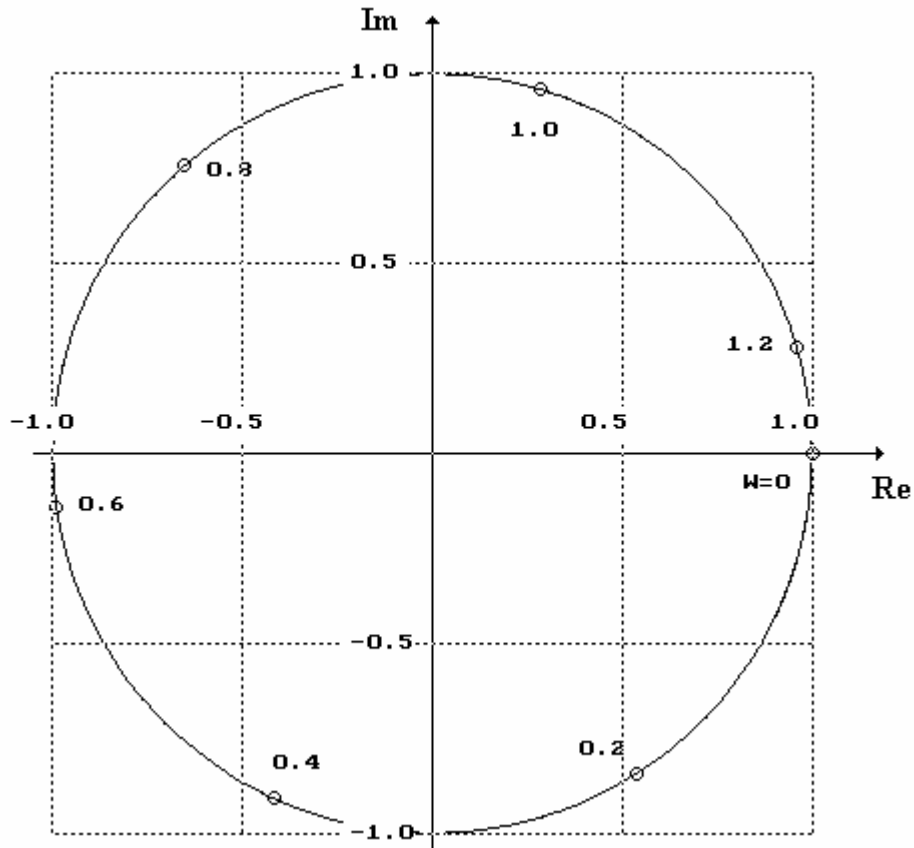


Рис. 1.25. Годограф амплитудно-фазовой характеристики звена транспортного запаздывания, $W(p) = e^{-5p}$.

1.8. Определение корней полинома для обратного преобразования Лапласа дробно-рациональных изображений

Математики переход от изображения Лапласа к оригиналу выполняют с помощью интеграла Бромвича (некоторые авторы называют его формулой Римана-Меллина)

$$f(t) = \frac{1}{2\pi j} \int_{s-j\infty}^{s+j\infty} \bar{f}(p) e^{pt} dt. \quad (1.108)$$

Здесь $s > s_0$, где s_0 - абсцисса сходимости.

Инженеры отдают предпочтение таблицам соответствий. Именно так и мы с вами определяли оригиналы для переходных характеристик элементарных динамических звеньев.

Если передаточная функция объекта является дробно-рациональной и изображение входного сигнала имеет форму отношения полиномов (а рассмотренные нами входные сигналы являются именно такими), то изображение выходного сигнала также будет иметь вид отношения полиномов. Общий подход к выполнению обратного преобразования в подобной ситуации такой. Изображение, если оно не представлено в таблице соответствий, разлагается на элементарные слагаемые, каждому из которых отвечает какой-то элемент таблицы. После определения коэффициентов разложения остается найти оригиналы для каждого из элементарных слагаемых. Сумма этих оригиналов и будет искомым оригиналом.

Структура слагаемых, на которые разлагается дробно-рациональное изображение, определяется корнями знаменателя этого изображения. Итак, первая проблема – это определение корней полинома знаменателя. Для полиномов выше второй степени корни приходится определять на компьютере численными методами, причем важно, чтобы метод безотказно срабатывал при любых степенях полинома и при произвольных наборах его коэффициентов. Он должен быстро сходиться и не зависеть от начальных точек поиска (начальных приближений).

Рассмотрим один из таких методов, который базируется на использовании критерия Гурвица, через определение степени стойкости. Такой подход абсолютно естественный для специалиста по автоматизации, поэтому мы его и выбрали. Понятия устойчивости и степени устойчивости динамической системы дальше будут рассматриваться более основательно, пока же достаточно оговорить, что устойчивость ассоциируется с отсутствием корней полинома в правой половине комплексной плоскости, а степень устойчивости – это расстояние от самого „правого” корня полинома до мнимой оси (степень устойчивости считается положительной, если исследуемый полином „устойчив” и отрицательной в противном случае).

Рассмотрим коротко этапы определения корней полинома. Пусть это будет полином $A(p)$, который задается массивом $A:\text{Coef}$. Соответствующие подпрограммы определены в модуле `Polynom`, листинг которого приводится в [6, 10].

Определение радиуса R окружности с центром в начале координат комплексной плоскости, которая охватывает все корни полинома по формуле

$$R = 1 + \frac{b}{|a_n|}, \quad (1.109)$$

где $b = \max(|a_0|, |a_1|, \dots, |a_{n-1}|)$. Реализуется формула (1.109) в подпрограмме-функции, заголовок которой

```
function Radius(A:Coef):real.
```

Оценка „устойчивости” полинома по критерию Гурвица [6, 9] – подпрограмма-функция с заголовком

```
function Hurvic(A:Coef):integer.
```

Функция `Hurvic` возвращает значения +1, если исследуемый полином является „устойчивым” и -1, если он неустойчив или находится на границе устойчивости.

Замена в полиноме $A(p)$ переменной p на $p_1 - L$, где L – некоторая константа.

$$B(p_1) = A(p) \Big|_{p=p_1-L}. \quad (1.110)$$

Коэффициенты полинома $B(p_1)$ вычисляются по формулам

$$B_s = \frac{1}{s!} \frac{d^s A(p)}{dp^s} \Big|_{p=-L}, \quad 0 \leq s \leq n. \quad (1.111)$$

Формулу (1.111) реализует подпрограмма `ZamPPL`, заголовок которой выглядит так

```
procedure ZamPPL(A:Coef; L:real; var B:Coef).
```

Определение степени устойчивости – подпрограмма `StepUst` с заголовком

```
function StepUst(A:Coef; Eps:real):real.
```

Параметр `Eps:real` – это допустимая погрешность в определении степени устойчивости. Суть алгоритма, реализованного в `StepUst`, состоит в том, что начальная зона поиска степени устойчивости считается ограниченной диапазоном $-R..R$, где R соответствует (1.109). Методом половинного деления с использованием подпрограмм `ZamPPL` (смещает мнимую ось в центр зоны поиска) и `Hurvic` (оценивает устойчивость „смещенного” полинома $B(p_1)$) зона поиска суживается до тех пор, пока ее ширина не становится меньше 2ε (параметр ε – это `Eps`). После этого центр такой “суженной” зоны точка p принимается за искомую точку, куда надо перенести мнимую ось чтобы „смещенный” полином вышел на границу устойчивости. При этом „новая” мнимая ось пройдет (с погрешностью не больше ε) через самый „правый” корень полинома $A(p)$ (в старой системе координат – базовой). Степень устойчивости принимается равной $-p$.

Определение крайнего правого корня реализуется в подпрограмме `KorPolSU`

```
procedure KorPolSU(A:Coef; Eps:real; var R, I:real).
```

Здесь `Eps` как и раньше допустимая погрешность в определении корней, R , I – вещественная и мнимая части самого правого корня.

В этой процедуре сначала с помощью подпрограммы `StepUst`

определяется R . После этого сканируется участок вертикали, которая проходит через точку R (имеет абсциссу R), начиная от $I = 0$, до $I = \sqrt{Rad^2 - R^2}$ на границе окружности радиуса R (функция Radius). В процессе сканирования с шагом H ищется точка, где модуль значения полинома $A(p)$ принимает наименьшее значение. Начальное значение шага H принимается равным одной десятой ширины зоны поиска. После локализации точки минимума зона поиска суживается: нижняя граница берется на шаг H ниже от найденной точки, а верхняя – на шаг H выше. Шаг уменьшается вдвое и поиск оптимальной точки продолжается, пока сканирование очередной зоны не будет проведено с шагом, который не превышает Eps . Найденная точка принимается за искомое значение корня полинома $A(p)$.

Определение массива корней полинома. Если найденный корень имеет мнимую часть I , модуль которой не превышает Eps , корень считается вещественным (I обнуляется), иначе корень рассматривается как комплексный. Поскольку полином $A(p)$ имеет вещественные коэффициенты, то наличие комплексного корня свидетельствует о существовании сопряженного ему корня, то есть оказывается, что найдено фактически два корня $(R + jI)$ и $(R - jI)$.

Чтобы найденный корень или пара сопряженных корней не мешали искать остальные корни полинома (а среди них могут быть и кратные корни, которые „прячутся” под найденными), надо найденные корни исключить из исследуемого полинома. Для этого надо $A(p)$ поделить на $(p - R)$ или соответственно на $(p - R - jI)(p - R + jI) = p^2 - 2Rp + R^2 + I^2$.

Эта операция реализуется подпрограммой DelPol, вызов которой выглядит так

$$\text{DelPol}(A, E, A, E),$$

где E – массив коэффициентов полинома $(p - R)$ или $(p^2 - 2Rp + R^2 + I^2)$ – смотри выше. Первый параметр A соответствует полиному $A(p)$ – из которого исключается полином $E(p)$, третий же параметр A – это полином-частное, то есть „редуцированный” полином с исключенными корнями (только что

найденными). Найденные корни $(R + jI)$ и $(R - jI)$ заносятся в массив `Korni:Kor`, где `Type Kor=array [1..2, -1..30] of real`. В первой строке – вещественные части корней, под ними во второй строке – мнимые.

С редуцированным полиномом поступаем точно так же, как с его предшественником, мы и назвали его точно так же - $A(p)$. И так продолжается до тех пор, пока степень полинома $A(p)$ не станет меньше или равной 2. С таким „ослабленным” полиномом может справиться уже и процедура `Kvadrat`, которая специализируется на вычислении корней полиномов первой или второй степени по известным формулам линейного или квадратного уравнения.

Эта процедура „доукомплектовывает” массив `Korni:Kor`. В ячейку `Korni[1, -1]` заносится номер последнего занесенного в него корня полинома, n – степень полинома $A(p)$, корни которого несет массив `Korni`.

Особенностью рассмотренного алгоритма есть то, что в нем корни определяются „справа-налево”. В результате кратные корни будут расположены в массиве подряд. Это позволяет просто объединять их в группы кратных, что важно при разложении изображения на элементарные слагаемые. За комплексным корнем автоматически располагается сопряженный ему корень. Это также удобно.

Отрицательной чертой алгоритма является накопление погрешностей. Чем “позже” определяется корень, тем, в принципе, большую погрешность несет он в себе. Существуют способы уменьшить эту погрешность, но авторы алгоритма не делали этого чтобы не усложнять алгоритм. А потом, погрешности можно уменьшить, но их нельзя устранить полностью. Они будут накапливаться все равно – больше или меньше. Так вот, положительной стороной рассмотренного алгоритма есть то, что точность определения корня тем выше, чем правее лежит корень. При исследовании автоматических систем именно „правые” корни определяют существенные особенности переходного процесса. Итак, метод рационально подходит к последовательности определения корней: сначала с наибольшей точностью определяются самые

“нужные” корни, а потом по мере послабления требований к точности определяется „второстепенные” корни.

Упомянутая выше особенность – накопление погрешностей – должна учитываться при задании ϵ_{ps} . Фактические погрешности при определении корней могут на порядки превышать заданное (задекларированное) значение допустимой погрешности. Читатель может убедиться в существовании подобного эффекта, предлагая компьютеру найти корни для полинома с известными корнями, в частности, с корнями высокой кратности. Накопление погрешностей может приводить к ошибкам в определении структуры найденных корней (простые они или кратные, вещественные или комплексные). А это, в свою очередь, может быть причиной серьезных неприятностей при выполнении обратного преобразования Лапласа. Будем возвращаться к этому обстоятельству при рассмотрении очередных шагов на пути компьютерного формирования формул оригиналов.

1.9. Объединение массивов корней полиномов и редактирование результирующего массива

Если исследуемое изображение можно представить в виде произведения каких-то дробно-рациональных сомножителей, то имеет смысл корни знаменателей каждого из этих сомножителей определять по отдельности. При этом не накапливаются погрешности, как это было при определении корней знаменателя, полученного путем перемножения сомножителей. Кроме того, суммарная трудоемкость определения корней полиномов-сомножителей с относительно невысокими степенями будет значительно меньше, чем трудоемкость определения корней полинома-произведения. Массивы корней для отдельных полиномов-сомножителей можно объединить в результирующий массив с помощью процедуры `DopMs`

```
procedure DopMs (Ms1,Ms2:Kor; var Ms:Kor);
  var N1,N2,N,S:integer;
begin
  N1:=round(Ms1[1,-1]); N2:=round(Ms2[1,-1]);
  Ms:=Ms1;
  for s:1 to N2 do
    begin
      Ms[1,N1+s]:=Ms2[1,s];
      Ms[2,N1+s]:=Ms2[2,s];
    end;
  Ms[1,-1]:=N1+N2
end;
```

Прежде чем разлагать результирующее дробно-рациональное изображение на элементарные слагаемые, надо объединить одинаковые корни его знаменателя в группы кратных корней. Если отдельные корни совпадают между собою абсолютно или отличаются один от другого существенным

образом, объединение в группы кратных корней не является проблемой. Проблема возникает тогда, когда корни близки между собой, но не совпадают. Тогда надо решать, надо ли их объединить в группу и если так, то вокруг которого из близких между собою корней или, возможно, надо как-то усреднять их. А может, следует оставить все как есть и рассматривать эти корни как простые? К сожалению, последний вариант часто оказывается неприемлемым, поскольку при близких между собою корнях соответствующие коэффициенты в разложении изображения на элементарные слагаемые, как правило, оказываются очень большими по модулю. Происходит это из-за того, что в знаменателях формул для коэффициентов присутствуют множители типа разностей между соседними корнями. Тогда, когда близость между корнями есть следствием погрешностей вычислений (при отсутствии которых, например, корни должны были бы быть точно кратными, то разность, на которую приходится делить, может быть по модулю близкой к погрешности – результат деления становится абсурдным, это фактически деление на нуль). Конечно, объединение близких (но не тождественных) корней в группу – это также зло, но не такое большое, как отказ от этого объединения. Другое дело – критерий. Где граница, которая определяет какие корни необходимо объединять, а какие нет? И еще одно обстоятельство. Как показывает опыт работы, например, с процедурой `KORPOLMS`, при определении вещественных корней относительно высокой кратности по крайней мере часть корней представляется в виде комплексно-сопряженных пар с небольшой по модулю мнимой частью. Комплексным парам, как известно, отвечают колебательные составляющие в переходных процессах. Если система устойчива (для неустойчивых процессов точность их расчета, как правило, несущественна – “отрубив голову – по волосам не плачут”), то колебания затухают, и при маленькой частоте колебаний (а модуль мнимой части корней является угловой частотой колебаний) эти составляющие успевают затухнуть, прежде чем их “колебательная сущность” успевает проявиться. В таких случаях, очевидно, целесообразно с самого начала игнорировать наличие “слабой” мнимой части

корня и рассматривать подобную сопряженную пару как пару одинаковых вещественных корней.

Корни, собранные в группы кратных, будем размещать в массиве типа `Mk`, где первые две строки будут такими же, как в массивах типа `Kor` (то есть будут содержать соответственно вещественную и мнимую части корней), а вот в третьей строке под каждым из корней будет записываться его кратность. Начальная кратность каждого из корней считается равной 1.

В подпрограмме `RedMask` комплексный корень рассматривается как вещественной когда модуль его мнимой части не превышает $\frac{1}{k_i}$ модуля его вещественной части. k_i - наперед заданное число. Корни считаются кратными, если расстояние между ними не превышает $\frac{1}{k}$ их среднего модуля. k - заданный коэффициент (k и k_i - настроечные коэффициенты, пользователь может задавать их по своему усмотрению).

```

procedure RedMask (M0:Kor; K,Ki:integer; var Ms:Mk);
var N0,S,Z,Z1:integer;
begin
  N0:=round(M0[1,-1]); Ms[1,-1]:=N0;
  for S:=1 to N0 do
    begin
      Ms[1,S]:=M0[1,S];
      Ms[3,S]:=1;
      if abs(M0[2,S])*ki<abs(M0[1,S])
        then Ms[2,S]:=0
        else Ms[2,S]:=M0[2,S]
    end;
  if N0>1 then
    begin
      S:=1;
      repeat
        if abs(Ms[2,S])>0 then Z:=S+2
          else Z:=S+1;
        while Z<=N0 do
          if sqrt(sqr(Ms[1,Z]-Ms[1,S])+sqr(Ms[2,Z]-
            Ms[2,S]))*k<sqrt(sqr(Ms[1,Z]

```

```

        +Ms[1,S])/2)+sqr((Ms[2,Z]+Ms[2,S])/2))
then
  begin
    Ms[3,S]:=Ms[3,S]+Ms[3,Z];
    if abs(Ms[2,Z])>0
    then
      begin
        dec(N0,2); Ms[3,S+1]:=Ms[3,S];
        for Z1:=Z to N0 do
          begin
            Ms[1,Z1]:=Ms[1,Z1+2];
            Ms[2,Z1]:=Ms[2,Z1+2];
            Ms[3,Z1]:=Ms[3,Z1+2]
          end
        end
      else
        begin
          dec(N0);
          for Z1:=Z to N0 do
            begin
              Ms[1,Z1]:=Ms[1,Z1+1];
              Ms[2,Z1]:=Ms[2,Z1+1];
              Ms[3,Z1]:=Ms[3,Z1+1]
            end
          end;
          Ms[1,-1]:=N0
        end
      else
        if abs(Ms[2,Z])>0 then inc(Z,2)
          else inc(Z);
        if abs(Ms[2,S])>0
        then
          begin
            Ms[3,S+1]:=Ms[3,S];
            inc(S,2)
          end
        else inc(s)
      until S>=N0
    end
  end;
end;

```

1.10. Алгебраические операции над полиномами с комплексными коэффициентами

При вычислении коэффициентов разложения изображения на элементарные слагаемые при наличии комплексных корней в знаменателе изображения коэффициенты разложения будут комплексными числами и ряд операций придется выполнять над полиномами с комплексными коэффициентами. Начнем с вычисления значений полинома, когда его коэффициенты – комплексные числа и аргумент также комплексный.

Будем сохранять значение коэффициентов полинома в массивах типа `Kor`. В массивах этого типа мы размещали значения корней. Договоримся, что информация о полиноме $A(p)$ с комплексными коэффициентами размещается в массиве `A:Kor` таким образом: в `A[1, -1]` – степень полинома, в `A[1, s]`, `A[2, s]` – вещественная и мнимая части s -го коэффициента (при p^s).

Вычисление значения полинома по схеме Хорнера при вещественных коэффициентах и комплексном значении аргумента реализуется процедурой `HorComp` [4, 6, 9]. Этот алгоритм используется в процедуре `HorCompC`, просто при добавлении к промежуточному произведению очередного коэффициента приходится в отдельности прибавлять его вещественную и мнимую части.

```
procedure HorCompC (A:Kor; R, I:real; var Re, Im:real);
var R1:real; n, s:integer;
begin
  N:=round(A[1, -1]);
  Re:=A[1, n]; Im:=A[2, n];
  for s:=N-1 downto 0 do
    begin
      R1:=Re*R-Im*I+A[1, s];
      Im:=Re*I+Im*R+A[2, s];
      Re:=R1
    end;
  end;
```

Алгебраическая сумма полиномов с комплексными коэффициентами отличается от аналогичной операции, которая реализуется в процедуре

Alsumpol для полиномов с вещественными коэффициентами лишь тем, что прибавлять коэффициенты надо как комплексные числа (в отдельности вещественные, в отдельности – мнимые части).

Для полиномов с комплексными коэффициентами это будет процедура SumpolC.

```

procedure SumpolC (A,B:Kor; Op:char; var C:Kor);
var Na, Nb, Nc, s:integer;
begin
  Na:=round(A[1,-1]);
  Nb:=round(B[1,-1]);
  if Na>Nb then Nc:=Na
    else Nc:=Nb;
  C[1,-1]:=Nc;
  for S:=0 to Nc do
    begin
      C[1,S]:=0; C[2,S]:=0;
      If s<=Na then
        begin
          C[1,S]:= C[1,S]+A[1,S];
          C[2,S]:= C[2,S]+A[2,S]
        end;
      If s<=Nb then
        If Op='+'
          then
            begin
              C[1,S]:= C[1,S]+B[1,S];
              C[2,S]:= C[2,S]+B[2,S]
            end
          else
            begin
              C[1,S]:= C[1,S]-B[1,S];
              C[2,S]:= C[2,S]-B[2,S]
            end
        end
    end
end;

```

При умножении полиномов с комплексными коэффициентами z-ый коэффициент полинома $A(p)$ множится на s-ий коэффициент полинома $B(p)$, то есть

$$(A_{1,z} + jA_{2,z})(B_{1,s} + jB_{2,s}) = C_{1,z+s} + jC_{2,z+s},$$

где $C_{1,z+s} = A_{1,z}B_{1,s} - A_{2,z}B_{2,s}$, $C_{2,z+s} = A_{1,z}B_{2,s} + A_{2,z}B_{1,s}$.

```

procedure UmnPolC (A,B:Kor; var C:Kor);
var Z, S, NA, NB, Nc:integer;
begin
  Na:=round(A[1,-1]); Nb:=round(B[1,-1]); Nc:=Na+Nb;
  C[1,-1]:=Nc;
  for S:=0 to Nc do
    begin
      C[1,s]:=0; C[2,s]:=0
    end;
  for Z:=0 to Na do
    for S:=0 to Nb do
      begin
        C[1,Z+S]:=C[1,Z+S]+A[1,Z]*B[1,S]-A[2,Z]*B[2,S];
        C[2,Z+S]:=C[2,Z+S]+A[1,Z]*B[2,S]+A[2,Z]*B[1,S]
      end
    end;
end;

```

Деление полиномов по схеме

$$\frac{A(p)}{B(p)} = C(p) + \frac{D(p)}{B(p)} \quad (1.112)$$

выполняет процедура DelPol. Умножим (1.112) на $B(p)$

$$A(p) = C(p) \cdot B(p) + D(p). \quad (1.113)$$

Пусть, для конкретности $Na = 6$, $Nb = 2$, тогда $Nc = Na - Nb = 4$, $Nd := Nb - 1$.

Разворачиваем формулу (1.113)

$$\begin{aligned} (C_6p^6 + C_5p^5 + C_4p^4 + C_3p^3 + C_2p^2 + C_1p^1 + C_0)(B_2p^2 + B_1p^1 + B_0) + D_1p + D_0 = \\ = A_6p^6 + A_5p^5 + A_4p^4 + A_3p^3 + A_2p^2 + A_1p^1 + A_0. \end{aligned}$$

Раскрываем скобки и приравниваем коэффициенты при одинаковых степенях p (все коэффициенты – комплексные, первый индекс 1 – для вещественных частей, 2 – для мнимых).

p^6	$C_{1,6}B_{1,0} - C_{2,6}B_{2,0} + C_{1,5}B_{1,1} - C_{2,5}B_{2,1} + C_{1,4}B_{1,2} - C_{2,4}B_{2,2} = A_{1,6}$
	$C_{1,6}B_{2,0} + C_{2,6}B_{1,0} + C_{1,5}B_{2,1} + C_{2,5}B_{1,1} + C_{1,4}B_{2,2} + C_{2,4}B_{1,2} = A_{2,6}$
p^5	$C_{1,5}B_{1,0} - C_{2,5}B_{2,0} + C_{1,4}B_{1,1} - C_{2,4}B_{2,1} + C_{1,3}B_{1,2} - C_{2,3}B_{2,2} = A_{1,5}$
	$C_{1,5}B_{2,0} + C_{2,5}B_{1,0} + C_{1,4}B_{2,1} + C_{2,4}B_{1,1} + C_{1,3}B_{2,2} + C_{2,3}B_{1,2} = A_{2,5}$
...	...
p^2	$C_{1,2}B_{1,0} - C_{2,2}B_{2,0} + C_{1,1}B_{1,1} - C_{2,1}B_{2,1} + C_{1,0}B_{1,2} - C_{2,0}B_{2,2} = A_{1,2}$
	$C_{1,2}B_{2,0} + C_{2,2}B_{1,0} + C_{1,1}B_{2,1} + C_{2,1}B_{1,1} + C_{1,0}B_{2,2} + C_{2,0}B_{1,2} = A_{2,2}$
p^1	$C_{1,1}B_{1,0} - C_{2,1}B_{2,0} + C_{1,0}B_{1,1} - C_{2,0}B_{2,1} + D_{1,1} = A_{1,1}$
	$C_{1,1}B_{2,0} + C_{2,1}B_{1,0} + C_{1,0}B_{2,1} + C_{2,0}B_{1,1} + D_{2,1} = A_{2,1}$
p^0	$C_{1,0}B_{1,0} - C_{2,0}B_{2,0} + D_{1,0} = A_{1,0}$
	$C_{1,0}B_{2,0} + C_{2,0}B_{1,0} + D_{2,0} = A_{2,0}$

Коэффициенты полинома-частного $C(p)$ с $Nc+1 \leq Z \leq Na$ считаются равными нулю ($C_{1,z} = 0$, $C_{2,z} = 0$).

Из первой пары уравнений для $Nc = 6$, которая рассматривается как система, определяется $C_{1,Nc}$, $C_{2,Nc}$, из второй - $C_{1,Nc-1}$, $C_{2,Nc-1}$ и так до $C_{1,0}$ и $C_{2,0}$. Как обобщение этого алгоритма можно записать для полинома-частного рекуррентные формулы

$$\begin{cases} C_{1,z} = \frac{(A_{1,Nb+Z} - S_{1,Z})B_{1,Nb} + (A_{2,Nb+Z} - S_{2,Z})B_{2,Nb}}{B_{1,Nb}^2 + B_{2,Nb}^2}, \\ C_{2,z} = \frac{(A_{2,Nb+Z} - S_{2,Z})B_{1,Nb} + (A_{1,Nb+Z} - S_{1,Z})B_{2,Nb}}{B_{1,Nb}^2 + B_{2,Nb}^2}, \end{cases} \quad (1.114)$$

где Z изменяется от Nc до 0,

$$S_{1,Z} = \sum_{s=0}^{Nb-1} (B_{1,s}C_{1,Nb+Z-s} - B_{2,s}C_{2,Nb+Z-s}),$$

$$S_{2,Z} = \sum_{s=0}^{Nb-1} (B_{1,s}C_{2,Nb+Z-s} - B_{2,s}C_{1,Nb+Z-s}).$$

Для полинома-остатка

$$\begin{cases} D_{1,Z} = A_{1,Z} - \sum_{S=0}^Z (B_{1,S}C_{1,Z-S} - B_{2,S}C_{2,Z-S}); \\ D_{2,Z} = A_{2,Z} - \sum_{S=0}^Z (B_{1,S}C_{2,Z-S} + B_{2,S}C_{1,Z-S}). \end{cases} \quad (1.115)$$

Формулы (1.114) и (1.115) реализует процедура DelpolC.

```

procedure Delpolc (A,B:Kor; var C,D:Kor);
var Z,S,Na,Nb,Nc,Nd,n:integer;
    Zn,S1,S2:real;
begin
  Na:=round(A[1,-1]); Nb:=round(B[1,-1]);
  if Na<Nb
  then
    begin
      D:=A; C[1,-1]:=0; C[1,0]:=0; C[2,0]:=0
    end
  else
    begin
      Nc:=Na-Nb; Nd:=Nb-1;
      C[1,-1]:=Nc; D[1,-1]:=Nd;
      for z:=Nc+1 to Na do
        begin
          C[1,z]:=0; C[2,z]:=0
        end;
      Zn:=sqr(B[1,Nb])+sqr(B[2,Nb]);
      for Z:=Nc downto 0 do
        begin
          S1:=0; S2:=0;
          for S:=0 to Nd do
            begin
              n:=Nb+Z-S;
              S1:=S1+B[1,S]*C[1,n]-B[2,S]*C[2,n];
              S2:=S2+B[1,S]*C[2,n]+B[2,S]*C[1,n]
            end;
          n:=Nb+Z;
          C[1,Z]:=((A[1,n]-S1)*B[1,Nb]+(A[2,n]-
            S2)*B[2,Nb])/Zn;
          C[2,Z]:=((A[2,n]-S2)*B[1,Nb]-(A[1,n]-
            S1)*B[2,Nb])/Zn
        end;
      for Z:=0 to Nd do
        begin
          S1:=0; S2:=0;

```

```
for S:=0 to Z do
  begin
    n:=Z-S;
    S1:=S1+B[1,S]*C[1,n]-B[2,S]*C[2,n];
    S2:=S2+B[1,S]*C[2,n]+B[2,S]*C[1,n]
  end;
  D[1,Z]:=A[1,Z]-S1;
  D[2,Z]:=A[2,Z]-S2
end
end
end;
```

1.11 Определение коэффициентов разложения дробно-рационального изображения на элементарные слагаемые

Итак, пусть имеем изображение

$$\bar{y}(p) = \frac{B(p)}{A(p)}, \quad (1.115)$$

где полином $A(p)$ имеет корни, содержащиеся в массиве $M_s : M_k$. И пусть среди этих корней есть корень p_s (вещественный или комплексный) кратности n_s . Этой группе корней в разложении (1.115) будет соответствовать группа элементарных слагаемых типа:

$$E_s(p) = \frac{D_{1,S}}{(p-p_s)^{n_s}} + \frac{D_{2,S}}{(p-p_s)^{n_s-1}} + \dots + \frac{D_{n_s-1,S}}{(p-p_s)^2} + \frac{D_{n_s,S}}{(p-p_s)}. \quad (1.116)$$

Коэффициенты в пределах группы будем определять последовательно, начиная с $D_{1,S}$. Пусть, например, речь идет о $D_{z,S}$, $1 \leq z \leq n_s - 1$. Согласно (1.115):

$$\frac{B_z C(p)}{(p-p_s)^{n_s+1-z}} \equiv \frac{D_{z,S}}{(p-p_s)^{n_s+1-z}} + \frac{D_{z+1,S}}{(p-p_s)^{n_s-z}} + \dots + \frac{D_{n_s,S}}{(p-p_s)^{n_s}} + R, \quad (1.117)$$

где $C(p)$ в знаменателе левой части $C(p) = \frac{A(p)}{(p-p_s)^{n_s}}$, а R – сумма элементарных слагаемых, которые соответствуют остальным корням $A(p)$ (исключая рассматриваемую s -ую группу).

Множим (1.117) на $(p-p_s)^{n_s+1-z}$ и рассматриваем полученный результат при $p = p_s$. Получаем:

$$D_{z,S} = \left. \frac{B_z(p)}{C(p)} \right|_{p=p_s}. \quad (1.118)$$

Поскольку p_s может быть как вещественным, так и комплексным числом, то и полиномы $B_z(p)$ и $C(p)$ в общем случае – полиномы с комплексными коэффициентами. Их значения при $p = p_s$ вычисляем с помощью процедуры `HorCompC`, которая возвращает для числителя

$B_z(p_s) = R_1 + jI_1$, а для знаменателя соответственно $C(p_s) = R_2 + jI_2$.

Результатом будет:

$$D_{z,s} = \frac{R_1 + jI_1}{R_2 + jI_2} = R + jI, \quad (1.119)$$

$$R = \frac{R_1 R_2 + I_1 I_2}{R_2^2 + I_2^2};$$

где

$$I = \frac{R_2 I_1 - R_1 I_2}{R_2^2 + I_2^2}.$$

После того как $D_{z,s}$ определено, слагаемое с этим коэффициентом в (1.117) переносится в левую часть тождества, после чего (1.117) приобретает вид:

$$\frac{B_z}{(p-p_s)^{n_s+1-z} C(p)} - \frac{D_{z,s}}{(p-p_s)^{n_s+1-z}} \equiv \frac{D_{z+1,s}}{(p-p_s)^{n_s-z}} + \dots + \frac{D_{n_s,s}}{(p-p_s)} + R. \quad (1.120)$$

Слагаемые в левой части (1.120) приводятся к общему знаменателю $(p-p_s)^{n_s+1-z} C(p)$, общий числитель левой части будет:

$$B_z(p) - C(p)D_{z,s}.$$

Этот числитель должен делиться без остатка на $(p-p_s)$, ведь правая часть (1.120) имеет в общем знаменателе $(p-p_s)^{n_s-z}$, то есть в степени на единицу меньше чем в левой части. Это возможно лишь тогда, когда и числитель и знаменатель левой части имеют общий множитель $(p-p_s)$, на который левая часть должна сокращаться. После упрощения левой части (1.120) приобретает вид:

$$\frac{B_{z+1}(p)}{(p-p_s)^{n_s+1-z} C(p)} \equiv \frac{D_{z+1,s}}{(p-p_s)^{n_s-z}} + \dots + \frac{D_{n_s,s}}{(p-p_s)} + R. \quad (1.121)$$

Из (1.21) $D_{z+1,s}$ определяется точно так же как $D_{z,s}$ с (1.120). Слагаемое, с найденным $D_{z+1,s}$ переносится влево... и так до тех пор, пока не будут определены все коэффициенты s -ой группы.

Если p_s – комплексная величина, то и коэффициенты $D_{z,s}$ будут

комплексными. Можно показать, что комплексным сопряженным корням (а такие корни обязательно размещаются подряд в массиве Ms) отвечают и сопряженные группы коэффициентов. Так, после группы вида (1.116) будет идти группа:

$$\frac{D_{z,s+1}}{(p-p_{s+1})^{n_s}} + \frac{D_{2,s+1}}{(p-p_{s+1})^{n_s-1}} + \dots + \frac{D_{n_{s-1},s+1}}{(p-p_{s+1})^2} + \frac{D_{n_s,s+1}}{(p-p_{s+1})} \quad (1.122)$$

где p_{s+1} - число, сопряженное p_s , например, для $p_s = \alpha_s + j\beta_s$ это $p_{s+1} = \alpha - j\beta_s$.

Итак, для $D_{z,s} = R_{z,s} + jI_{z,s}$, это $D_{z,s+1} = R_{z,s} - jI_{z,s}$. С учетом этого будем записывать в массив Dz комплексные значения коэффициентов $D_{z,s}$ так: вещественную часть $R_{z,s}$ в $Dz[z, s]$, а мнимую $I_{z,s}$ - в $Dz[z, s+1]$.

```

procedure Dzs(Br:Coef; Ms:Mk; var Dz:Mkr);
{ type Mk=array[1..3,-1..10] of real;
  Mkr=array[1..10,1..10] of real;
  Ms[1,S], Ms[2,S] - вещественная и мнимая части
  s-ой группы, Ms[3,S]=Ns-кратность s-ой группы
  Ms[-1,1]=Ng - число групп }
var Ns,Ng,Z,S,J:integer;
    A,B,C,D,E,F:Kor;
    R1,I1,R2,I2,Zn,R,I:real;
begin
  B[1,-1]:Br[1,-1];
  for S:=0 to round (Br[1,-1]) do
    begin
      B[1,S]:Br[S]; B[2,S]:=0
    end;
  Ng:=round(Ms[1,-1]);
  A[1,-1]:=0; A[1,0]:=1; A[2,0]:=0;
  E[1,-1]:=1; E[1,1]:=1; E[2,1]:=0;
  for s:=1 to Ng do
    begin
      Ns:=round(Ms[3,S]); E[1,0]:=-Ms[1,S];
      E[2,0]:=-Ms[2,S];
      for Z:=1 to Ns do UmnPol(A,E,A)
    end;
  S:=1;
  repeat
    Ns:=round(Ms[3,S]);
    E[1,0]:=-Ms[1,S]; E[2,0]:=-abs(Ms[2,S]);

```

```

D:=A; C:=B;
for j:=1 to Ns do DelPolC(D,E,D,F);
HorCompC(D,Ms[1,S],Ms[2,S],R2,I2);
Zn:=sqr(R2)+sqr(I2);
for Z:=1 to Ns do
begin
HorCompC(C,Ms[1,S],Ms[2,S],R1,I1);
if Zn=0
then
begin
R:=0; I:=0
end
else
begin
R:=(R1*R2+I1*I2)/Zn;
I:=(R2*I1-R1*I2)/Zn
end;
Dz[Z,S]:R;
if Ms[2,S]<>0 then Dz[Z,S+1]:=I;
if Z<Ns then
begin
F[1,-1]:=0; F[1,0]:R; F[2,0]:=I;
UmnPol(D,F,F0; Sumpoll(C,F,' ',C);
DelPolC(C,E,C,F)
end
end;
if Ms[2,S]=0 then inc(s) else inc(s,2)
until S>Ng
end;
end;

```


2. Аналитические модели простейших технологических объектов и элементов систем управления

Для компьютерного моделирования автоматических систем исследователь должен иметь в своем распоряжении информацию о динамических свойствах всех элементов, входящих в состав этой системы. Будем называть совокупность данных, характеризующих особенности динамического поведения элемента системы (или системы в целом) его математической моделью. Существует ряд способов задания динамических характеристик: система дифференциальных уравнений, переходная или импульсная функция, частотные характеристики в разных формах и т.п. Выбор того или иного способа определяется конкретной ситуацией: способом получения модели, решаемой задачей, методами и средствами решения задачи и т.д. вплоть до личных предпочтений исследователя. Как правило, между различными формами представления динамических характеристик существует связь, позволяющая пересчитывать их одни в другие. Так что в данном отношении выбор исходной формы не является определяющим.

Однако среди перечисленных форм представления динамических свойств элемента системы следует выделить первую, а именно систему дифференциальных уравнений. Подобные модели получаются, как правило, аналитическим путем и предполагает глубокое понимание (с соответствующим отображением в модели) исследователем сути физических процессов, протекающих в изучаемом объекте (системе). В виде дифференциальных уравнений отображаются (в зависимости от типа объекта) законы сохранения вещества и (или) энергии, законы физики, химии, термодинамики, механики и т.д. Получаемая таким образом математическая модель отображает структуру объекта, учитывает механизмы взаимодействия между собой отдельных элементов. С ее помощью можно анализировать влияние на свойства объекта отдельных факторов как-то: внешних воздействий, конструктивных и режимных параметров, физических и технологических ограничений и других обстоятельств. Это позволяет целенаправленно изменять характеристики

объекта в желаемом направлении.

Аналитическая модель описывает целый ряд однотипных объектов, она легко позволяет учитывать эффект масштабирования, модификации в конструкции, изменения в характеристиках объекта при переходе из одного режима в другой, от одной нагрузки к другой. И еще одна очень существенная особенность аналитической модели в виде системы дифференциальных уравнений – она отображает динамические свойства объекта в целом, определяет динамику всех возможных каналов объекта в их взаимозависимости. Она также позволяет естественным образом задавать (учитывать) начальные и граничные условия, учитывать технические и режимные ограничения, учитывать возможные нелинейности.

Практически всем физическим объектам присуща пространственная распределенность параметров (все явления разворачиваются в пространстве и времени). Но математические модели, отображающие эффект распределенности в пространстве, являются относительно сложными, что, естественно, усложняет как их создание, так и дальнейшее использование. С целью упрощения пытаются когда только это представляется возможным упростить модель, сводя объект к конечному набору аккумулирующих емкостей, каждая из которых рассматривается как элемент с сосредоточенными параметрами. Подобный подход как раз и используется в настоящем пособии.

Аналитическое моделирование динамического поведения технологических аппаратов, установок предполагает глубокое проникновение в суть процессов, протекающих в изучаемом объекте. Это, как правило, под силу только специалисту в соответствующей области. В противном случае модель, предложенная дилетантом, может оказаться сколь угодно далекой от истины. Но с чего-то начинать надо. В данной главе рассматривается ряд простейших объектов, для составления математических моделей которых достаточно самых элементарных сведений из школьного курса физики. Авторы просто демонстрируют читателю общие подходы к составлению данного класса моделей и формы их представления. Компьютерные алгоритмы обработки этих моделей – в следующей главе.

2.1. Математическая модель многоемкостного теплового объекта типа проточной емкости

Рассмотрим проточную емкость рис. 2.1, через которую протекает теплоноситель в жидком состоянии (жидкость мало изменяет свой объем при изменении давления).

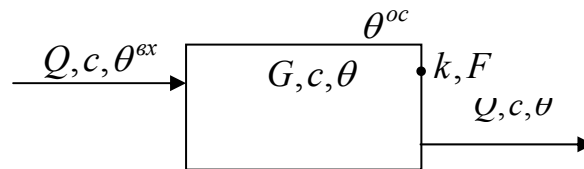


Рис. 2.1. Расчетная схема проточной тепловой емкости.

Если объем емкости считать неизменным в процессе работы и расход теплоносителя, который входит в емкость, равняется расходу, который выходит из нее, то уравнение элементарного теплового баланса (для промежутка времени dt) такой емкости можно в первом приближении записать так

$$\left[Qc\theta^{ex} - Qc\theta - kF(\theta - \theta^{oc}) \right] dt = d(Gc\theta), \quad (2.1)$$

здесь Q - расход теплоносителя через емкость, c - коэффициент теплоемкости теплоносителя, θ^{ex} - температура на входе в емкость, θ - температура в емкости, θ^{oc} - температура окружающей среды, k - коэффициент теплопередачи между теплоносителем в емкости и окружающей средой, F - поверхность теплообмена между теплоносителем и средой, t - время, dt - дифференциал (бесконечно малое приращение) времени, G - количество теплоносителя, $d(Gc\theta)$ - дифференциал от количества тепла в емкости. В квадратных скобках левой части (2.1) приведен баланс прихода и расхода тепла в емкости в единицу времени. $Qc\theta^{ex}$ - тепло, которое вносится в емкость с потоком теплоносителя, $Qc\theta$ - тепло, которое выносится из емкости с потоком теплоносителя, $kF(\theta - \theta^{oc})$ - тепло, которое теряет емкость за счет теплообмена со средой, в которой находится емкость.

Обращаем внимание читателя на условность выбора направления движения тепла за счет теплообмена с окружающей средой. Тепло само „лучше

знает” куда ему направляться – из емкости в окружающую среду или наоборот: тепло всегда “перетекает” из мест с большей температурой в места, где она ниже.

Предполагая, что тепло “оставляет” емкость, мы в выражении $kF(\theta - \theta^{oc})$ из температуры θ вычитаем θ^{oc} , если бы мы предположили противоположное направление теплового потока, то надо было бы знаки в скобках изменить на противоположные. В результате составная теплового баланса из $-kF(\theta - \theta^{oc})$ изменилась бы на $+kF(\theta^{oc} - \theta)$, то есть фактически осталась бы такой же.

На самом деле мы можем фактически и не знать настоящего направления движения тепла. В процессе изменения состояния объекта могут изменяться θ и θ^{oc} , может изменяться знак их разности. При этом уравнение (2.1) остается справедливым.

При записи уравнения (2.1) кроме вышеупомянутых сделан ряд предположений. Относительно самой емкости принята модель так называемого идеального смешения. Считаем, что теплоноситель мгновенно и идеально перемешивается, в результате чего его температура усредняется по массе теплоносителя. Это температура θ (именно с этой температурой теплоноситель покидает емкость). Эта же температура определяет тепловой поток через стенки емкости. Очевидно, что для перемешивания надо расходовать энергию и чем интенсивнее перемешивание, тем, очевидно, больше. Но эта составляющая энергии (а энергия на перемешивание превращается в тепловую энергию - в тепло) в уравнении (2.1) не отображена. Это будет справедливо когда данная составляющая мала сравнительно с учтенными в (2.1) тепловыми потоками. Не учтены также кинетические энергии потоков, которые входят и выходят из емкости. Не учтена зависимость величины коэффициента теплоемкости c от температуры. Оправданием может служить относительно слабая зависимость теплоемкости жидкости от ее температуры в относительно узком диапазоне изменений температуры. Последнее обстоятельство (малый диапазон ожидаемых изменений) мотивируется назначением математической модели

объекта для исследования его поведения в составе системы автоматического управления, в которой регулятор должен сам „позаботиться” о том, чтобы температура мало отклонялась от ее заданного значения, в окрестности которого модель объекта будет адекватной.

Относительно расчетного базового режима должны определяться и значения параметров $Q, c, G, \theta^{oc}, \theta^{ex}$, которые принимают участие в (2.1) как коэффициенты. При изменении режима, то есть при изменении отдельных или всех перечисленных коэффициентов динамические свойства будут изменяться. В различных режимах объект может вести себя совсем по-разному.

Поделим (1.1) на dt и перенесем в левую часть слагаемые с θ и его производной, а в правую часть – остальное. Получим

$$Gc \frac{d\theta}{dt} + (Qc + kF)\theta = Qc\theta^{ex} + kF\theta^{oc}. \quad (2.2)$$

Мы считали, что G и c от времени не зависят.

В дифференциальном уравнении (2.2) можно сделать один из коэффициентов равным единице (поделив все члены уравнения на него). Пусть это будет коэффициент при θ

$$T \frac{d\theta}{dt} + \theta = k_{ex}\theta^{ex} + k_{oc}\theta^{oc}, \quad (2.3)$$

где $T = \frac{Gc}{Qc + kF}$; $k_{ex} = \frac{Qc}{Qc + kF}$; $k_{oc} = \frac{kF}{Qc + kF} = 1 - k_{ex}$.

Возможен и другой вариант

$$\frac{d\theta}{dt} = \frac{Qc\theta^{ex} + kF\theta^{oc} - (Qc + kF)\theta}{Gc}. \quad (2.4)$$

Это так называемая каноническая форма. Мы будем широко пользоваться ею в дальнейшем.

Объект, который описывается дифференциальным уравнением (2.3) или (2.4), можно рассматривать как имеющий два возможных входных сигнала (θ^{ex} и θ^{oc}) и один выходной сигнал (θ).

Как уже упоминалось раньше, в автоматике принято называть входными сигналами (входами) возможные причины изменения состояния (это, как правило, внешние факторы), тогда как выходами считаются параметры состояния самого объекта.

Итак, структурно рассматриваемый объект можно представить так

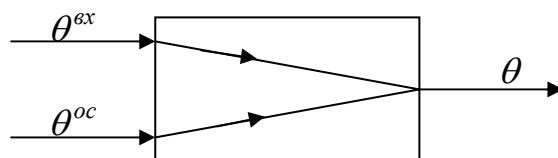


Рис.2.2 Структурная схема проточной емкости.

Если определенный входной сигнал влияет на рассматриваемый выход, то говорят, что между данным входом и выходом существует канал передачи сигнала. На структурной схеме в границах прямоугольника, который изображает объект, прямые линии показывают отдельные каналы (часто со стрелками). Каждый из каналов может быть описан отдельным дифференциальным уравнением. Эти уравнения можно получить, например, из (2.3) таким образом.

Начнем с того, что понятия о каналах и их динамических свойствах (независимых от свойств других каналов объекта) в первую очередь ориентированы на стационарные (не изменяющие своих свойств во времени) и линейные объекты.

Линейными называют объекты, реакция которых на сумму входных сигналов равняется сумме реакций на каждый из входных сигналов в отдельности. Эта особенность линейных объектов отвечает так называемому принципу суперпозиции (наложения).

Рассмотрим (2.3) в статическом (базовом) режиме. Статический режим означает, что состояние объекта с течением времени не изменяется, итак, все производные по времени от всех входных и выходных сигналов в объекте равняются нулю. Дифференциальное уравнение (2.3) в статическом режиме

приобретает вид

$$\theta^* = k_{\text{ex}}\theta^{\text{ex}*} + k_{\text{oc}}\theta^{\text{oc}*}, \quad (2.5)$$

здесь индекс * указывает на статическое значение соответствующей переменной.

Представим теперь в (2.3) переменные таким образом

$$\begin{cases} \theta = \theta^* + \Delta\theta, \\ \theta^{\text{ex}} = \theta^{\text{ex}*} + \Delta\theta^{\text{ex}}, \\ \theta^{\text{oc}} = \theta^{\text{oc}*} + \Delta\theta^{\text{oc}}. \end{cases} \quad (2.6)$$

здесь $\Delta\theta, \Delta\theta^{\text{ex}}, \Delta\theta^{\text{oc}}$ - приращения (относительно статических значений) переменных $\theta, \theta^{\text{ex}}, \theta^{\text{oc}}$. Подставим теперь (2.6) в (2.3)

$$T \frac{d\Delta\theta}{dt} + \theta^* + \Delta\theta = k_{\text{ex}}\theta^{\text{ex}*} + k_{\text{ex}}\Delta\theta^{\text{ex}} + k_{\text{oc}}\theta^{\text{oc}*} + k_{\text{oc}}\Delta\theta^{\text{oc}}. \quad (2.7)$$

Вычтем почленно уравнение (2.5) из (2.7). Получим

$$T \frac{d\Delta\theta}{dt} + \Delta\theta = k_{\text{ex}}\Delta\theta^{\text{ex}} + k_{\text{oc}}\Delta\theta^{\text{oc}}. \quad (2.8)$$

Об уравнение (2.8) говорят, что оно записано в приращениях переменных.

Обратим внимание читателя на то, что структура уравнения (2.8) оказалась такой же, как структура (2.3). Это вообще характерно для линейных дифференциальных уравнений. Линейное уравнение остается справедливым как в случае, когда речь идет об „абсолютных” значениях переменных, так и об их приращениях. Учитывая это обстоятельство мы в дальнейшем будем записывать линейные дифференциальные уравнения в форме (2.3), а будем иметь (по умолчанию) в виду форму (2.8), то есть формально будем опускать значок Δ в уравнениях относительно приращений.

Когда за начальное состояние объекта, описанного дифференциальным уравнением (2.8), принимается статический режим, то приращения его переменных $(\theta, \theta^{\text{ex}}, \theta^{\text{oc}})$ относительно этого статического режима будут, разумеется, нулевыми. Это и есть как раз так званые „нулевые начальные условия”, которыми мы широко будем пользоваться в дальнейшем.

А теперь вообразим, что, например, на каком-то интервале времени приращение $\Delta\theta^{oc} = 0$, тогда дифференциальное уравнение сводится к

$$T \frac{d\Delta\theta}{dt} + \Delta\theta = k_{ex} \Delta\theta^{ex}. \quad (2.9)$$

Аналогично при $\Delta\theta^{ex} = 0$ будем иметь

$$T \frac{d\Delta\theta}{dt} + \Delta\theta = k_{oc} \Delta\theta^{oc}. \quad (2.10)$$

Дифференциальное уравнение (2.9) описывает динамические свойства канала „ $\theta^{ex} \rightarrow \theta$ ”, тогда как (2.10) - канала „ $\theta^{oc} \rightarrow \theta$ ”. Канал указывается входным и выходным сигналами, между которыми (стрелкой) фиксируется причинно-следственная связь. Соответственно можно задать передаточные функции этих каналов

$$W_{\theta^{ex}, \theta}(p) = \frac{k_{ex}}{Tp + 1}, \quad (2.11)$$

$$W_{\theta^{oc}, \theta}(p) = \frac{k_{oc}}{Tp + 1}. \quad (2.12)$$

Усложним теперь математическую модель рассматриваемого объекта, учитывая теплоаккумулирующую способность стенок емкости (рис. 2.3).

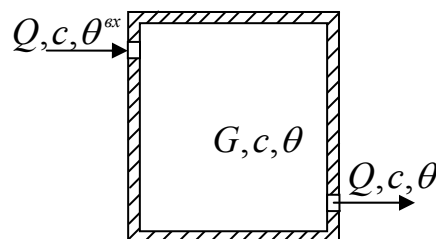


Рис. 2.3. Расчетная схема проточной емкости с учетом аккумуляции тепла в ее стенках.

Предположим, что теплоаккумулирующая способность стенок емкости заслуживает того, чтобы ее учли при моделировании теплового режима емкости. В предыдущем варианте мы (по умолчанию) считали, что теплоаккумулирующая способность стенок емкости настолько незначительна (по сравнению с теплоаккумулирующей способностью теплоносителя в емкости), что ею можно было пренебречь. Воздержимся пока что от

формулирования критерия, по которому можно оценивать степень существенности (или несущественности) той или иной аккумулирующей емкости в том или ином случае.

Итак, теперь, когда мы считаем, что стенка достойна того, чтобы ее рассматривать как „субъект” в процессе теплообмена, то уравнения элементарного теплового баланса надо будет составлять не только для теплоносителя (как это было в предыдущем случае), но и для стенки как отдельной тепловой емкости.

При такой постановке проблемы теплоноситель в емкости будет передавать тепло не непосредственно окружающей среде, как раньше, а стенке, разделяющей теплоноситель и среду. Стенка выступает в роли посредника. Коэффициент теплопередачи k разбивается на две составляющие, а именно на α_0 - коэффициент теплообмена между теплоносителем и стенкой и α_1 - между стенкой и средой. Очевидно, они связаны между собою известным из теплопередачи соотношением

$$k = \frac{1}{\frac{1}{\alpha_0} + \frac{\delta}{\lambda} + \frac{1}{\alpha_1}}.$$

Если мы будем рассматривать стенку как „тонкую”, то есть такую, в которой можно не учитывать распределенность температуры по ее толщине, то слагаемое $\frac{\delta}{\lambda}$ в знаменателе приведенной формулы можно опустить. Здесь δ - толщина стенки, λ - коэффициент теплопроводности материала стенки.

Стенка, в свою очередь, может обмениваться теплом не только с теплоносителем, но и с окружающей средой. Если поверхность теплообмена между теплоносителем и стенкой обозначим как F_0 , то площадь теплообмена между стенкой и средой будет F_1 (они могут отличаться одна от другой за счет конструктивных особенностей емкости).

Итак, вместо одного уравнения элементарного теплового баланса теперь будем иметь два

$$\begin{cases} Qc\theta^{6x} - Qc\theta - \alpha_0 F_0 (\theta - \theta^{cm}) = \frac{d}{dt}(Gc\theta), & (2.13) \\ \alpha_0 F_0 (\theta - \theta^{cm}) - \alpha_1 F_1 (\theta^{cm} - \theta^{oc}) = \frac{d}{dt}(G_{cm}c_{cm}\theta^{cm}). & (2.14) \end{cases}$$

Здесь $G_{cm}, c_{cm}, \theta^{cm}$ - соответственно масса стенки, коэффициент теплоемкости ее материала и усредненная температура стенки.

Считая, что G_{cm}, c_{cm} (как раньше G, c) не зависят от времени, систему (2.13) - (2.14) можно представить как

$$\begin{cases} Gc \frac{d\theta}{dt} + (Qc + \alpha_0 F_0)\theta = Qc\theta^{6x} + \alpha_0 F_0 \theta^{cm}, & (2.15) \\ G_{cm}c_{cm} \frac{d\theta^{cm}}{dt} + (\alpha_0 F_0 + \alpha_1 F_1)\theta^{cm} = \alpha_0 F_0 \theta + \alpha_1 F_1 \theta^{oc} & (2.16) \end{cases}$$

или так

$$\begin{cases} \frac{d\theta}{dt} = \frac{Qc\theta^{6x} - (Qc + \alpha_0 F_0)\theta + \alpha_0 F_0 \theta^{cm}}{Gc}, & (2.17) \\ \frac{d\theta^{cm}}{dt} = \frac{\alpha_1 F_1 \theta^{oc} + \alpha_0 F_0 \theta - (\alpha_0 F_0 + \alpha_1 F_1)\theta^{cm}}{G_{cm}c_{cm}}. & (2.18) \end{cases}$$

Это система уравнений в канонической форме.

Систему дифференциальных уравнений (2.15)–(2.16) можно свести к одному дифференциальному уравнению относительно выходной величины θ или, соответственно, θ^{cm} в зависимости от того, что нас больше интересует в данный момент времени.

Избавимся, например, в данной системе от θ^{cm} . Для этого определим θ^{cm} из (2.15)

$$\theta^{cm} = \frac{Gc \frac{d\theta}{dt} + (Qc + \alpha_0 F_0)\theta - Qc\theta^{6x}}{\alpha_0 F_0}. \quad (2.19)$$

Подставим теперь (2.19) в (2.16)

$$\frac{G_{cm}c_{cm} \left[Gc \frac{d^2\theta}{dt^2} + (Qc + \alpha_0 F_0) \frac{d\theta}{dt} - Qc \frac{d\theta^{ex}}{dt} \right]}{\alpha_0 F_0} +$$

$$+ \frac{(\alpha_0 F_0 + \alpha_1 F_1) \left[Gc \frac{d\theta}{dt} + (Qc + \alpha_0 F_0) \theta - Qc \theta^{ex} \right]}{\alpha_0 F_0} = \alpha_0 F_0 \theta + \alpha_1 F_1 \theta^{oc}.$$

Множим на $\alpha_0 F_0$, раскрываем скобки и сводим подобные

$$GcG_{cm}c_{cm} \frac{d^2\theta}{dt^2} + [G_{cm}c_{cm}(Qc + \alpha_0 F_0) + Gc(\alpha_0 F_0 + \alpha_1 F_1)] \frac{d\theta}{dt} +$$

$$[(\alpha_0 F_0 + \alpha_1 F_1)(Qc + \alpha_0 F_0) - (\alpha_0 F_0)^2] \theta = \quad (2.20)$$

$$= Qc \left[G_{cm}c_{cm} \frac{d\theta^{ex}}{dt} + (\alpha_0 F_0 + \alpha_1 F_1) \theta^{ex} \right] + \alpha_1 F_1 \alpha_0 F_0 \theta^{oc}.$$

Вводим обозначения

$$a_2 = GcG_{cm}c_{cm}; \quad a_1 = G_{cm}c_{cm}(Qc + \alpha_0 F_0) + Gc(\alpha_0 F_0 + \alpha_1 F_1);$$

$$a_0 = (\alpha_0 F_0 + \alpha_1 F_1)(Qc + \alpha_0 F_0) - (\alpha_0 F_0)^2;$$

$$b_1^{ex,0} = QcG_{cm}c_{cm}; \quad b_0^{ex,0} = Qc(\alpha_0 F_0 + \alpha_1 F_1); \quad b_0^{oc,0} = \alpha_1 F_1 \alpha_0 F_0.$$

Тогда (2.20) приобретает вид

$$a_2 \frac{d^2\theta}{dt^2} + a_1 \frac{d\theta}{dt} + a_0 \theta = b_1^{ex,0} \frac{d\theta^{ex}}{dt} + b_0^{ex,0} \theta^{ex} + b_0^{oc,0} \theta^{oc}. \quad (2.21)$$

Избавимся теперь в системе (2.15)-(2.16) от переменной θ . Для этого определим ее из (2.16)

$$\theta = \frac{G_{cm}c_{cm} \frac{d\theta^{cm}}{dt} + (\alpha_0 F_0 + \alpha_1 F_1) \theta^{cm} - \alpha_1 F_1 \theta^{oc}}{\alpha_0 F_0}$$

и подставим в (1.15)

$$Gc \left[\frac{G_{cm}c_{cm} \frac{d^2\theta^{cm}}{dt^2} + (\alpha_0 F_0 + \alpha_1 F_1) \frac{d\theta^{cm}}{dt} - \alpha_1 F_1 \frac{d\theta^{oc}}{dt}}{(\alpha_0 F_0)} + \right.$$

$$\left. + \frac{(Qc + \alpha_0 F_0) \left[G_{cm}c_{cm} \frac{d\theta^{cm}}{dt} + (\alpha_0 F_0 + \alpha_1 F_1) \theta^{cm} - \alpha_1 F_1 \theta^{oc} \right]}{(\alpha_0 F_0)} \right] = Qc \theta^{ex} + \alpha_0 F_0 \theta^{cm}$$

Упрощаем

$$\begin{aligned}
 & GcG_{cm}c_{cm} \frac{d^2\theta^{cm}}{dt^2} + [G_{cm}c_{cm}(Qc + \alpha_0F_0) + Gc(\alpha_0F_0 + \alpha_1F_1)] \frac{d\theta^{cm}}{dt} + \\
 & + [(\alpha_0F_0 + \alpha_1F_1)(Qc + \alpha_0F_0) - (\alpha_0F_0)^2] \theta^{cm} = \\
 & = \alpha_1F_1 \left[Gc \frac{d\theta^{oc}}{dt} + (Qc + \alpha_0F_0)\theta^{oc} \right] + Qc\alpha_0F_0\theta^{ex}.
 \end{aligned} \tag{2.22}$$

Учитывая предварительно введенные обозначения (a_2, a_1, a_0) и вводя новые

$$b_1^{oc,1} = \alpha_1F_1Gc; \quad b_0^{oc,1} = \alpha_1F_1(Qc + \alpha_0F_0); \quad b_0^{ex,1} = Qc\alpha_0F_0,$$

переписываем (1.22) в виде

$$a_2 \frac{d^2\theta^{cm}}{dt^2} + a_1 \frac{d\theta^{cm}}{dt} + a_0\theta^{cm} = b_1^{oc,1} \frac{d\theta^{oc}}{dt} + b_0^{oc,1}\theta^{oc} + b_0^{ex,1}\theta^{ex}. \tag{2.23}$$

Акцентируем внимание читателя на том обстоятельстве, что структура и коэффициенты левых частей дифференциальных уравнений (2.21) и (2.23) идентичны. Практически всегда существует возможность привести эти уравнения к виду, когда эта идентичность будет иметь место. Она обусловлена тем, что оба дифференциальные уравнения описывают один и тот же физический объект. Левые части дифференциальных уравнений отображают свойства этого объекта, присущие всем его динамическим каналам, тогда как правые части дифференциальных уравнений отображают специфику каждого из каналов. Структурную схему данного объекта на базе уравнений (2.21), (2.23) можно представить так

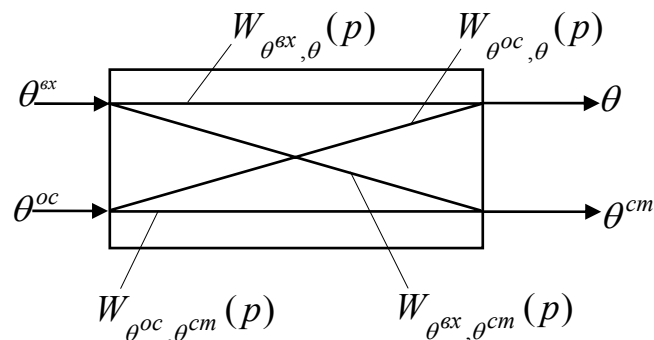


Рис. 2.4. Структурная схема объекта, изображенного на рис. 2.3.

Сами же передаточные функции будут

$$W_{\theta^{ex},\theta}(p) = \frac{b_1^{ex,0} p + b_0^{ex,0}}{a_2 p^2 + a_1 p + 1}, \quad (2.24)$$

$$W_{\theta^{oc},\theta}(p) = \frac{b_0^{oc,0}}{a_2 p^2 + a_1 p + 1}, \quad (2.25)$$

$$W_{\theta^{oc},\theta^{cm}}(p) = \frac{b_1^{oc,1} p + b_0^{oc,1}}{a_2 p^2 + a_1 p + 1}, \quad (2.26)$$

$$W_{\theta^{ex},\theta^{cm}}(p) = \frac{b_0^{ex,1}}{a_2 p^2 + a_1 p + 1}. \quad (2.27)$$

Все передаточные функции (2.24) – (2.27) имеют общий знаменатель (общие „фамильные” черты).

Если теперь дополнительно принять во внимание еще теплоаккумулирующую способность, например, мешалки массой G_m (если мешалка, конечно, есть и ее масса $G_m \neq 0$), теплоемкостью c_m , поверхностью теплообмена F_m (с теплоносителем) и коэффициентом теплоотдачи α_m , то система элементарных балансов может выглядеть так

$$\left\{ \begin{array}{l} Qc\theta^{ex} - Qc\theta - \alpha_0 F_0 (\theta - \theta^{cm}) - \alpha_m F_m (\theta - \theta^m) = \frac{d}{dt} (Gc\theta), \end{array} \right. \quad (2.28)$$

$$\left\{ \begin{array}{l} \alpha_0 F_0 (\theta - \theta^{cm}) - \alpha_1 F_1 (\theta^{cm} - \theta^{oc}) = \frac{d}{dt} (G_{cm} c_{cm} \theta^{cm}), \end{array} \right. \quad (2.29)$$

$$\left\{ \begin{array}{l} \alpha_m F_m (\theta - \theta^m) = \frac{d}{dt} (G_m c_m \theta^m). \end{array} \right. \quad (2.30)$$

Здесь θ^m - усредненная температура мешалки.

Систему (2.28) – (2.30) можно также представить в канонической форме

$$\left\{ \begin{array}{l} \frac{d\theta}{dt} = \frac{Qc\theta^{ex} - (Qc + \alpha_0 F_0 + \alpha_m F_m)\theta + \alpha_0 F_0 \theta^{cm} + \alpha_m F_m \theta^m}{(Gc)}, \end{array} \right. \quad (2.31)$$

$$\left\{ \begin{array}{l} \frac{d\theta^{cm}}{dt} = \frac{\alpha_0 F_0 \theta - \alpha_1 F_1 \theta^{oc} - (\alpha_0 F_0 + \alpha_1 F_1) \theta^{cm}}{(G_{cm} c_{cm})}, \end{array} \right. \quad (2.32)$$

$$\left\{ \begin{array}{l} \frac{d\theta^m}{dt} = \frac{\alpha_m F_m \theta - \alpha_m F_m \theta^m}{(G_m c_m)}. \end{array} \right. \quad (2.33)$$

Вывести дифференциальные уравнения и (или) передаточные функции отдельных каналов предлагаем читателю самостоятельно. В третьем разделе

данного пособия мы рассмотрим компьютерный алгоритм определения передаточных функций всех возможных каналов объекта, который задан системой n обыкновенных дифференциальных уравнений в канонической форме.

2.2. Моделирования объектов с регулированием давления пара или газа

Пусть объект исследования имеет структуру, показанную на рис. 2.5.

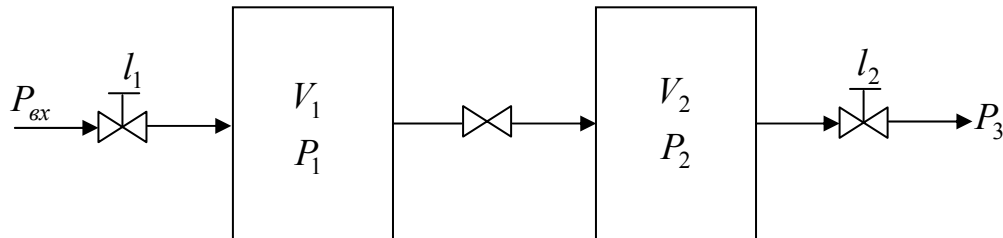


Рис. 2.5. Расчетная схема двухкорпусного ресивера.

Итак, имеем две газовые емкости, объемы которых V_1 и V_2 не изменяются в процессе работы (не зависят от давлений P_1 да P_2 и от времени). Запишем уравнения элементарных материальных балансов для каждой из емкостей.

$$\left\{ \begin{array}{l} k_1 l_1 \sqrt{(P_{ex} - P_1)P_{ex}} - k \sqrt{(P_1 - P_2)P_1} = \frac{d}{dt}(V_1 \rho_1), \\ k \sqrt{(P_1 - P_2)P_1} - k_2 l_2 \sqrt{(P_2 - P_3)P_2} = \frac{d}{dt}(V_2 \rho_2). \end{array} \right. \quad (2.34)$$

$$\left\{ \begin{array}{l} k_1 l_1 \sqrt{(P_{ex} - P_1)P_{ex}} - k \sqrt{(P_1 - P_2)P_1} = \frac{d}{dt}(V_1 \rho_1), \\ k \sqrt{(P_1 - P_2)P_1} - k_2 l_2 \sqrt{(P_2 - P_3)P_2} = \frac{d}{dt}(V_2 \rho_2). \end{array} \right. \quad (2.35)$$

Здесь k, k_1, k_2 - коэффициенты; l_1, l_2 - степени открытия клапанов (в абсолютных или относительных величинах); P_{ex}, P_1, P_2, P_3 - соответственно давление газа на входе в ресивер, в 1-м корпусе, во 2-м корпусе и на выходе из ресивера; ρ_1, ρ_2 - плотность газа в 1-м и 2-м корпусах.

Плотность газа зависит от давления. Примем для простоты газ за идеальный $\frac{P}{\rho} = RT$, где R - газовая константа, T - абсолютная температура.

Расходы газа через клапаны записаны для ситуации, когда в процессе работы имеет место лишь докритический режим движения газа, то есть когда скорость газа как перед, так и после сужения остается значительно меньше скорости звука на соответствующем участке (при соответствующем давлении). В конкретном случае формулы для расчета расхода газа через сужение (клапан – это сужение) зависят от конструкции этого сужения, физических

характеристик газа, величин давления на входе и выходе, ускорения силы земного тяготения, и т.п. Для конкретности остановимся на приведенных формулах.

Будем рассматривать P_{ex}, l_1, l_2, P_3 как независимые входы, а P_1 и P_2 как выходы исследуемого объекта. Рассматривая, как выше отмечалось, V_1 и V_2 как константы. Решаем уравнения (2.34), (2.35) относительно производных

$$\begin{cases} \frac{dP_1}{dt} = f_1(l_1, P_{ex}, P_1, P_2, T), \\ \frac{dP_2}{dt} = f_2(P_1, P_2, l_2, P_3, T), \end{cases} \quad (2.36)$$

$$\quad (2.37)$$

где $f_1(l_1, P_{ex}, P_1, P_2, T) = \frac{RT}{V_1} [k_1 l_1 \sqrt{(P_{ex} - P_1)P_{ex}} - k \sqrt{(P_1 - P_2)P_1}]$,

$$f_2(P_1, P_2, l_2, P_3, T) = \frac{RT}{V_2} [k \sqrt{(P_1 - P_2)P_1} - k_2 l_2 \sqrt{(P_2 - P_3)P_2}].$$

Уравнения (2.34), (2.35) или, что одно и то же, (2.36), (2.37) нелинейные. Линеаризуем, например, (2.34) и (2.35) в окрестности статического режима, который принимается за базовый. Этот базовый режим определяется системой уравнений

$$\begin{cases} k_1 l_1^* \sqrt{(P_{ex}^* - P_1^*)P_{ex}^*} = k \sqrt{(P_1^* - P_2^*)P_1^*}, \\ k \sqrt{(P_1^* - P_2^*)P_1^*} = k_2 l_2^* \sqrt{(P_2^* - P_3^*)P_2^*}. \end{cases} \quad (2.38)$$

$$\quad (2.39)$$

Здесь символ * указывает на принадлежность переменной статическому режиму. Эта система уравнений в принципе может быть решена относительно P_1^* и P_2^* (в функции от $P_{ex}^*, l_1^*, l_2^*, P_3^*$).

Линеаризацию выполняем путем разложения каждого компонента системы в ряд Тейлора в окрестности статического режима с последующим вычитанием из полученного уравнения (2.38) или (2.39) для установившегося режима.

$$k_1 \sqrt{(P_{\text{ex}}^* - P_1^*) P_{\text{ex}}^*} \cdot \Delta l_1 + \frac{k_1 l_1^* \cdot 2P_{\text{ex}}^*}{2\sqrt{(P_{\text{ex}}^* - P_1^*) P_{\text{ex}}^*}} \Delta P_{\text{ex}} - \frac{k_1 l_1^* \cdot P_{\text{ex}}^*}{2\sqrt{(P_{\text{ex}}^* - P_1^*) P_{\text{ex}}^*}} \Delta P_1 -$$

$$- \frac{k \cdot 2P_1^*}{2\sqrt{(P_1^* - P_2^*) P_1^*}} \Delta P_1 + \frac{k P_1^*}{2\sqrt{(P_1^* - P_2^*) P_1^*}} \Delta P_2 = \frac{V_1}{RT} \frac{d\Delta P_1}{dt}. \quad (2.40)$$

$$\frac{k \cdot 2P_1^*}{2\sqrt{(P_1^* - P_2^*) P_1^*}} \Delta P_1 - \frac{k P_1^*}{2\sqrt{(P_1^* - P_2^*) P_1^*}} \Delta P_2 - k_2 \sqrt{(P_2^* - P_3^*) P_2^*} \Delta l_2 -$$

$$- \frac{k_2 l_2^* \cdot 2P_2^*}{2\sqrt{(P_2^* - P_3^*) P_2^*}} \Delta P_2 + \frac{k_2 l_2^* P_2^*}{2\sqrt{(P_2^* - P_3^*) P_2^*}} \Delta P_3 = \frac{V_2}{RT} \frac{d\Delta P_2}{dt}. \quad (2.41)$$

Систему дифференциальных уравнений (2.40), (2.41) перепишем в канонической форме

$$\left\{ \begin{aligned} \frac{dP_1}{dt} &= a_{1,1} \Delta P_1 + a_{1,2} \Delta P_2 + b_{1,1} \Delta P_{\text{ex}} + b_{1,2} \Delta l_1 + b_{1,3} \Delta l_2 + b_{1,4} \Delta P_3, \\ \frac{dP_2}{dt} &= a_{2,1} \Delta P_1 + a_{2,2} \Delta P_2 + b_{2,1} \Delta P_{\text{ex}} + b_{2,2} \Delta l_1 + b_{2,3} \Delta l_2 + b_{2,4} \Delta P_3, \end{aligned} \right. \quad (2.42)$$

$$\left\{ \begin{aligned} \frac{dP_1}{dt} &= a_{1,1} \Delta P_1 + a_{1,2} \Delta P_2 + b_{1,1} \Delta P_{\text{ex}} + b_{1,2} \Delta l_1 + b_{1,3} \Delta l_2 + b_{1,4} \Delta P_3, \\ \frac{dP_2}{dt} &= a_{2,1} \Delta P_1 + a_{2,2} \Delta P_2 + b_{2,1} \Delta P_{\text{ex}} + b_{2,2} \Delta l_1 + b_{2,3} \Delta l_2 + b_{2,4} \Delta P_3, \end{aligned} \right. \quad (2.43)$$

где $a_{1,1} = \frac{-RT \left[\frac{k_1 l_1^* \sqrt{P_{\text{ex}}^*}}{2\sqrt{P_{\text{ex}}^* - P_1^*}} + \frac{k \sqrt{P_1^*}}{\sqrt{P_1^* - P_2^*}} \right]}{V_1}$; $a_{1,2} = \frac{RTk \sqrt{P_1^*}}{2V_1 \sqrt{P_1^* - P_2^*}}$;

$$b_{1,1} = \frac{RTk_1 l_1^* \sqrt{P_{\text{ex}}^*}}{V_1 \sqrt{P_{\text{ex}}^* - P_1^*}}; \quad b_{1,2} = \frac{RTk_1 \sqrt{(P_{\text{ex}}^* - P_1^*) P_{\text{ex}}^*}}{V_1}; \quad b_{1,3} = 0; \quad b_{1,4} = 0;$$

$$a_{2,1} = \frac{RTk \sqrt{P_1^*}}{V_2 \sqrt{P_1^* - P_2^*}}; \quad a_{2,2} = -\frac{RT}{V_2} \left[\frac{k \sqrt{P_1^*}}{2\sqrt{P_1^* - P_2^*}} + \frac{k l_2^* \sqrt{P_2^*}}{\sqrt{P_2^* - P_3^*}} \right]; \quad b_{2,1} = 0; \quad b_{2,2} = 0;$$

$$b_{2,3} = -\frac{RT}{V_2} k_2 \sqrt{(P_2^* - P_3^*) P_2^*}; \quad b_{2,4} = \frac{RTk_2 l_2^*}{V_2} \cdot \frac{\sqrt{P_2^*}}{2\sqrt{P_2^* - P_3^*}};$$

Осталось только обозначить ΔP_1 как y_1 , ΔP_2 как y_2 , а ΔP_{ex} как x_1 , Δl_1 как x_2 , Δl_2 как x_3 и ΔP_3 как x_4 чтобы система (2.42), (2.43) приобрела целиком стандартный канонический вид у которой $n = 2$, $m = 4$

$$\left\{ \begin{array}{l} \frac{dy_1}{dt} = a_{1,1}y_1 + a_{1,2}y_2 + b_{1,1}x_1 + b_{1,2}x_2 + b_{1,3}x_3 + b_{1,4}x_4, \\ \frac{dy_2}{dt} = a_{2,1}y_1 + a_{2,2}y_2 + b_{2,1}x_1 + b_{2,2}x_2 + b_{2,3}x_3 + b_{2,4}x_4. \end{array} \right. \quad (2.44)$$

$$\left\{ \begin{array}{l} \frac{dy_1}{dt} = a_{1,1}y_1 + a_{1,2}y_2 + b_{1,1}x_1 + b_{1,2}x_2 + b_{1,3}x_3 + b_{1,4}x_4, \\ \frac{dy_2}{dt} = a_{2,1}y_1 + a_{2,2}y_2 + b_{2,1}x_1 + b_{2,2}x_2 + b_{2,3}x_3 + b_{2,4}x_4. \end{array} \right. \quad (2.45)$$

2.3. Моделирования объектов с регулированием уровня жидкости

Для конкретности рассмотрим такую расчетную схему

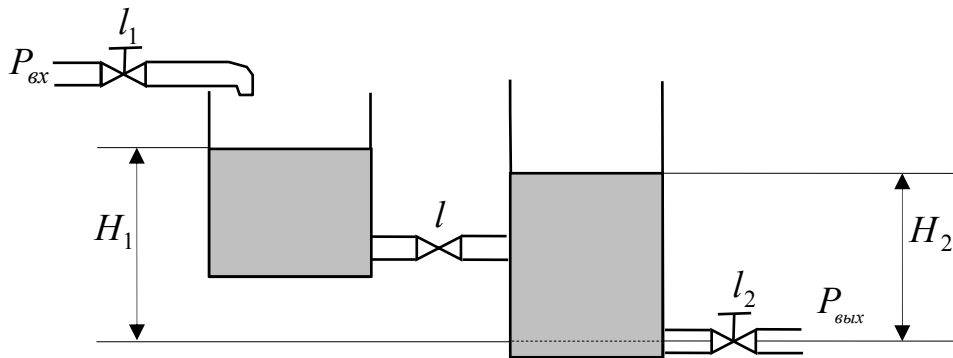


Рис. 2.6. Расчетная схема системы двух последовательных емкостей с регулированием уровня.

Запишем уравнение элементарных балансов для расхода жидкости в каждой из емкостей

$$\left\{ \begin{array}{l} k_1 l_1 \sqrt{P_{вх}} - k \sqrt{(H_1 - H_2) \rho} = \frac{d}{dt} (F_1 H_1), \\ k \sqrt{(H_1 - H_2) \rho} - k_2 l_2 \sqrt{H_2 \rho - P_{вых}} = \frac{d}{dt} (F_2 H_2). \end{array} \right. \quad (2.46)$$

$$\left\{ \begin{array}{l} k_1 l_1 \sqrt{P_{вх}} - k \sqrt{(H_1 - H_2) \rho} = \frac{d}{dt} (F_1 H_1), \\ k \sqrt{(H_1 - H_2) \rho} - k_2 l_2 \sqrt{H_2 \rho - P_{вых}} = \frac{d}{dt} (F_2 H_2). \end{array} \right. \quad (2.47)$$

Здесь k_1, k, k_2 - коэффициенты, которые зависят от конструкций клапанов и характеристик протекающей через них жидкости; l_1, l_2 - степени открытия клапанов на входе и выходе схемы; F_1, F_2 - площади поперечного сечения первой и второй емкостей; ρ - плотность жидкости; H_1, H_2 - уровни жидкости в емкостях, измеренные от общего базового значения (на уровне выходного клапана); $P_{вх}, P_{вых}$ - давления жидкости на входе и выходе схемы (измеренные от атмосферного).

Слагаемые в левых частях уравнений (2.46) и (2.47) – объемные расходы через клапаны. Предлагаем читателю самостоятельно выписать соответствующие уравнения для статического режима. Линеаризованные уравнения запишем так

$$\left\{ \begin{array}{l} k_1 \sqrt{P_{ex}^*} \Delta l_1 + \frac{k_1 l_1^*}{2\sqrt{P_{ex}^*}} \Delta P_{ex} - \frac{k}{2\sqrt{(H_1^* - H_2^*)\rho}} \Delta H_1 - \\ - \frac{k}{2\sqrt{(H_1^* - H_2^*)\rho}} \Delta H_2 = F_1 \frac{d\Delta H_1}{dt}, \end{array} \right. \quad (2.48)$$

$$\left\{ \begin{array}{l} \frac{k}{2\sqrt{(H_1^* - H_2^*)\rho}} \Delta H_1 - \frac{k}{2\sqrt{(H_1^* - H_2^*)\rho}} \Delta H_2 - k_2 \sqrt{H_2^* \rho - P_{eux}^*} \Delta l_2 - \\ - \frac{k_2 l_2^* \rho}{2\sqrt{H_2^* \rho - P_{eux}^*}} \Delta H_2 + \frac{k_2 l_2^*}{2\sqrt{H_2^* \rho - P_{eux}^*}} \Delta P_{eux} = F_2 \frac{d\Delta H_2}{dt}. \end{array} \right. \quad (2.49)$$

Читатель может самостоятельно привести эту систему дифференциальных уравнений к канонической форме.

2.4. Объекты с регулированием концентрации

Рассмотрим объект, смеситель двух потоков жидкости некоторого вещества, каждый из которых имеет концентрацию C_1 и C_2 соответственно. В смесителе, как это вытекает из самого названия, эти потоки смешиваются, концентрация усредняется (она обозначается как C). Считаем, что количество растворителя в смесителе остается все время неизменным, например, за счет того, что растворитель отбирается из емкости путем перелива. Расчетная схема такого растворителя показанная на рис.2.7.

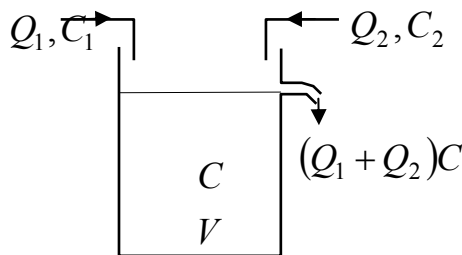


Рис.2.7. Расчетная схема смесителя.

Баланс вещества, концентрация которого моделируется,

$$Q_1 C_1 + Q_2 C_2 - (Q_1 + Q_2) C = \frac{d}{dt}(VC). \quad (2.50)$$

Здесь Q_1, Q_2 - объемные расходы растворителя в первом и втором входных потоках; C_1, C_2, C - объемные концентрации контролируемого вещества в двух входных потоках, непосредственно в смесителе и на его выходе соответственно; V - объем смесителя. Будем рассматривать Q_1, C_1, Q_2, C_2 как входные переменные, C - как выходную.

Линеаризируем (1.50)

$$C_1^* \Delta Q_1 + Q_1^* \Delta C_1 + C_2^* \Delta Q_2 + Q_2^* \Delta C_2 - C^* \Delta Q_1 - C^* \Delta Q_2 = V \frac{d\Delta C}{dt},$$

или после сведения подобных

$$\frac{d\Delta C}{dt} = a_{1,1} \Delta C + b_{1,1} \Delta Q_1 + b_{1,2} \Delta C_1 + b_{1,3} \Delta Q_2 + b_{1,4} \Delta C_2, \quad (2.51)$$

где $a_{1,1} = -\frac{C^*}{V}$; $b_{1,1} = \frac{(C_1^* - C^*)}{V}$; $b_{1,2} = \frac{Q_1^*}{V}$; $b_{1,3} = \frac{(C_2^* - C^*)}{V}$; $b_{1,4} = \frac{Q_2^*}{V}$.

Уравнения (1.51) – обыкновенное дифференциальное уравнение в канонической форме, в которой $n = 1$, $m = 4$.

2.5. Динамика механических систем

Как классический пример механической системы можно рассмотреть груз на пружине рис. 1.8.

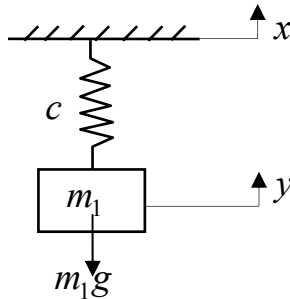


Рис. 2.8. Расчетная схема динамики массы на пружине

Пусть x - перемещение верхнего конца пружины жесткости c , y - перемещение ее нижнего конца, на котором зафиксирована масса m_1 . Будем рассматривать x как входной сигнал, y - как выходной. Массой пружины, сопротивлением воздуха при движении массы, а также трением в материале пружины при ее деформациях в первом приближении пренебрегаем. Рассмотрим условие динамического равновесия массы, на которую действует сверху сила деформации пружины, снизу – сила ее тяжести (за счет земного тяготения, g - величина ускорения земного тяготения). По принципу Даламбера к этим силам надо прибавить силу инерции $m_1 y''$, направленную в сторону, противоположную направлению ускорения. Итак, имеем

$$c(x - y) - m_1 g = m_1 y'' \quad (2.52)$$

При переходе к приращениям составляющая, соответствующая силе тяжести ($m_1 g$), исчезнет, поскольку мы считаем m_1 и g не зависящим от y . В результате для приращений будем иметь

$$m_1 \Delta y'' + c \Delta y = c \Delta x \quad (2.53)$$

Это дифференциальное уравнение можно представить в виде двух дифференциальных уравнений в канонической форме, например, так

$$\begin{cases} y_1' = y_2, & (2.54) \\ y_2' = \frac{c}{m_1}(-\Delta y_1 + \Delta x), & (2.55) \end{cases}$$

где y_1 - это y .

Если принять, что сопротивление движению массы m_1 пропорционально скорости ее движения, то дифференциальное уравнение (2.52) приобретет вид (в приращениях)

$$c(\Delta x - \Delta y) - k\Delta y' = m\Delta y''$$

или

$$m_1\Delta y'' + k\Delta y' + c\Delta y = c\Delta x$$

или, в канонической форме

$$\begin{cases} y_1' = a_{1,1}y_1 + a_{1,2}y_2 + b_{1,1}x_1, & (2.56) \\ y_2' = a_{2,1}y_1 + a_{2,2}y_2 + b_{2,1}x_1, & (2.57) \end{cases}$$

где $a_{1,1} = 0$; $a_{1,2} = 1$; $b_{1,1} = 0$; $a_{2,1} = -\frac{c}{m_1}$; $a_{2,2} = -\frac{k}{m_1}$; $b_{2,1} = \frac{c}{m_1}$.

Рассмотрим теперь математическую модель паровой турбины с двумя ступенями и одним отбором пара [8] – рис. 2.9.

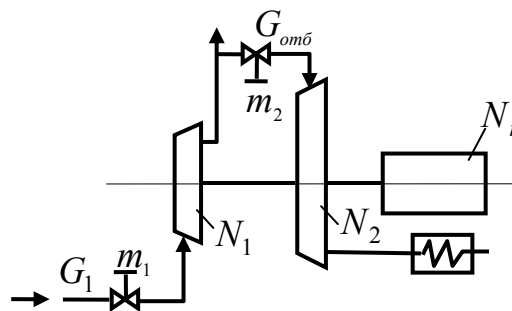


Рис. 2.9. Расчетная модель паровой турбины с отбором пара.

Запишем уравнение элементарного баланса энергии для турбины

$$J \frac{d\omega}{dt} = N_1 + N_2 - N_r, \quad (2.58)$$

где J - момент инерции системы ротор турбины-генератор; ω - угловая скорость вращения ротора; N_1 - мощность ступени высокого давления (1-ой

ступени); N_2 - мощность ступени низкого давления (2-ой ступени); N_r - мощность сопротивления, которое создает генератор.

Считаем, что

$$N_1 = f_1(m_1, \omega, P), \quad (2.59)$$

$$N_2 = f_2(m_2, P, \omega). \quad (2.60)$$

Здесь m_1, m_2 - степени открытия клапанов, которые регулируют расход пара в турбину (m_1) и подачу пара на 2-ую ступень (m_2); P - давление пара в камере отбора (на выходе из первой ступени).

Для камеры отбора элементарный материальный баланс пара

$$G - G_{отб} - G_2 = V \frac{d\rho}{dt}. \quad (2.61)$$

Здесь,

$$G = f_3(m_1, \omega, P), \quad (2.62)$$

$$G_2 = f_4(m_2, \omega, P), \quad (2.63)$$

ρ - плотность пара в камере отбора; V - объем камеры отбора.

Считаем, что процесс в камере отбора является политропическим с показателем политропы n , то есть описывается уравнением

$$\frac{P}{\rho^n} = const. \quad (2.64)$$

Линеаризованный вариант (1.64) имеет вид

$$\frac{(\rho^*)^n \Delta P - P^* n (\rho^*)^{n-1} \Delta \rho}{(\rho^*)^{2n}} = 0,$$

откуда

$$\Delta \rho = \frac{\rho^*}{nP^*}. \quad (2.65)$$

Линеаризируем (2.58) с учетом (2.59), (2.60), а также (2.61) с учетом (2.62), (2.63) и (2.65)

$$\begin{cases} \frac{dy_1}{dt} = a_{1,1}y_1 + a_{1,2}y_2 + b_{1,1}x_1 + b_{1,2}x_2 + b_{1,3}x_3 + b_{1,4}x_4, \\ \frac{dy_2}{dt} = a_{2,1}y_1 + a_{2,2}y_2 + b_{2,1}x_1 + b_{2,2}x_2 + b_{2,3}x_3 + b_{2,4}x_4. \end{cases} \quad (2.66)$$

$$\begin{cases} \frac{dy_1}{dt} = a_{1,1}y_1 + a_{1,2}y_2 + b_{1,1}x_1 + b_{1,2}x_2 + b_{1,3}x_3 + b_{1,4}x_4, \\ \frac{dy_2}{dt} = a_{2,1}y_1 + a_{2,2}y_2 + b_{2,1}x_1 + b_{2,2}x_2 + b_{2,3}x_3 + b_{2,4}x_4. \end{cases} \quad (2.67)$$

Здесь обозначено

$$y_1 = \Delta\omega; \quad y_2 = \Delta P;$$

$$x_1 = \Delta m_1; \quad x_2 = \Delta m_2; \quad x_3 = \Delta N_r; \quad x_4 = \Delta G_{\text{сид}};$$

$$a_{1,1} = \frac{\left(\frac{\partial f_1}{\partial \omega} - \frac{\partial f_2}{\partial \omega}\right)}{J}; \quad a_{1,2} = \frac{\left(\frac{\partial f_1}{\partial P} - \frac{\partial f_2}{\partial P}\right)}{J};$$

$$b_{1,1} = \frac{1}{J} \frac{\partial f_1}{\partial m_1}; \quad b_{1,2} = -\frac{1}{J} \frac{\partial f_2}{\partial m_2}; \quad b_{1,3} = -\frac{1}{J}; \quad b_{1,4} = 0;$$

$$a_{2,1} = \left(\frac{\partial f_3}{\partial \omega} - \frac{\partial f_4}{\partial \omega}\right) \frac{nP^*}{V\rho^*}; \quad a_{2,2} = \left(\frac{\partial f_3}{\partial P} - \frac{\partial f_4}{\partial P}\right) \frac{nP^*}{V\rho^*};$$

$$b_{2,1} = \frac{\partial f_3}{\partial m_1} \frac{nP^*}{V\rho^*}; \quad b_{2,2} = -\frac{\partial f_4}{\partial m_2} \frac{nP^*}{V\rho^*}; \quad b_{2,3} = 0; \quad b_{2,4} = -\frac{nP^*}{V\rho^*}.$$

Система (2.66)–(2.67) – обыкновенная система дифференциальных уравнений в канонической форме, где $n = 2$, $m = 4$.

2.6. RLC – цепочки

Рассмотрим, например, такую электрическую схему

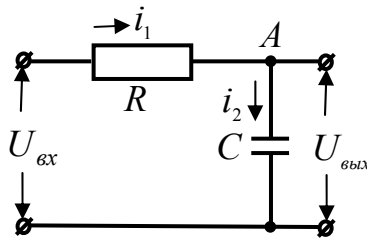


Рис.3.10. Расчетная схема RC – цепочки

Запишем баланс токов для узла в точке A , в соответствии с законом Кирхгофа

$$i_1 = i_2. \quad (2.68)$$

Уравнение (1.68) будет справедливым при условии, если прибор, который подключается к клеммам $U_{вых}$ имеет большое входное сопротивление и током, который протекает через это сопротивление (нагрузку), можно пренебречь (условие „высокоомности” нагрузки). Как известно

$$\left\{ \begin{aligned} i_1 &= \frac{U_{вх} - U_{вых}}{R}, \\ U_{вых} &= \frac{1}{C} \int_0^t i_2 dt. \end{aligned} \right. \quad (2.69)$$

$$\left\{ \begin{aligned} i_1 &= \frac{U_{вх} - U_{вых}}{R}, \\ U_{вых} &= \frac{1}{C} \int_0^t i_2 dt. \end{aligned} \right. \quad (2.70)$$

Дифференцируем (2.70) по t и заменяем в нем i_2 через i_1 согласно (2.68)

$$\frac{dU_{вых}}{dt} = \frac{1}{C} \left(\frac{U_{вх} - U_{вых}}{R} \right),$$

откуда

$$RC \frac{dU_{вых}}{dt} + U_{вых} = U_{вх} \quad (2.71)$$

или

$$\frac{dU_{вых}}{dt} = -\frac{1}{RC} U_{вых} - \frac{1}{RC} U_{вх}. \quad (2.72)$$

Пример LR – цепочки

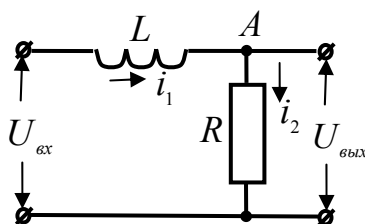


Рис. 2.11. Расчетная схема LR – цепочки

Уравнения (2.68) для точки A остается справедливым, но

$$\left\{ \begin{array}{l} L \frac{di_1}{dt} = U_{\text{вх}} - U_{\text{вых}} \end{array} \right. \quad (2.73)$$

$$\left\{ \begin{array}{l} i_2 = \frac{U_{\text{вых}}}{R} \end{array} \right. \quad (2.74)$$

Подставляем i_2 из (2.74) в (2.73) и учитывая (2.68), имеем

$$\frac{L}{R} \frac{dU_{\text{вых}}}{dt} + U_{\text{вых}} = U_{\text{вх}} \quad (2.75)$$

Дифференциальные уравнения (2.71) и (2.75) описывают апериодическое звено 1-го порядка.

Пусть теперь имеем такую схему

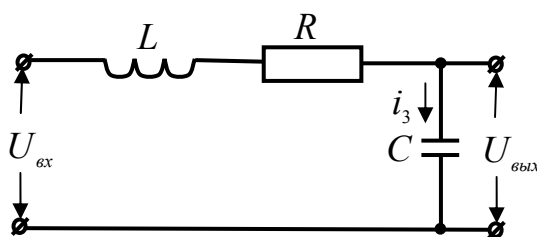


Рис. 2.12. Расчетная схема RLC – цепочки.

Очевидно, что

$$\left\{ \begin{array}{l} L \frac{di}{dt} + Ri = U_{\text{вх}} - U_{\text{вых}} \end{array} \right. \quad (2.76)$$

$$\left\{ \begin{array}{l} \frac{1}{C} \int_0^t i dt = U_{\text{вых}} \end{array} \right. \quad (2.77)$$

Дифференцируем (2.77) по t

$$i = C \frac{dU_{\text{вых}}}{dt},$$

и подставляем его в (2.76)

$$LC \frac{d^2 U_{\text{вых}}}{dt^2} + RC \frac{dU_{\text{вых}}}{dt} + U_{\text{вых}} = U_{\text{ex}}. \quad (2.78)$$

При условии что

$$R^2 C < 4L \quad (2.79)$$

это будет колебательное звено. При невыполнении условия (2.79) – апериодическое звено 2-го порядка.

И еще один пример

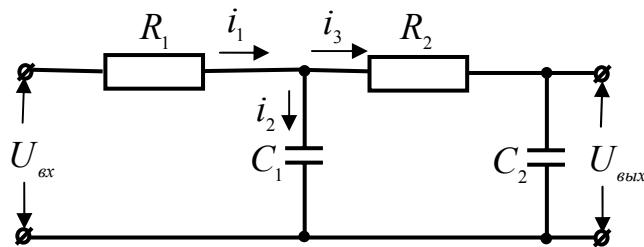


Рис. 2.13. Расчетная схема двухемкостной RC – цепочки.

Легко видеть что

$$\left\{ \begin{array}{l} R_1 i_1 + \frac{1}{C_1} \int_0^t i_2 dt = U_{\text{ex}}, \end{array} \right. \quad (2.80)$$

$$\left\{ \begin{array}{l} i_1 = i_2 + i_3, \end{array} \right. \quad (2.81)$$

$$\left\{ \begin{array}{l} R_1 i_1 + R_2 i_3 = U_{\text{ex}} - U_{\text{вых}}, \end{array} \right. \quad (2.82)$$

$$\left\{ \begin{array}{l} \frac{1}{C_2} \int_0^t i_3 dt = U_{\text{вых}}. \end{array} \right. \quad (2.83)$$

Из (2.82) после дифференцирования

$$i_3 = C_2 \frac{dU_{\text{вых}}}{dt}. \quad (2.84)$$

Из (2.82) с учетом (2.84)

$$i_1 = \frac{1}{R_1} (U_{\text{ex}} - U_{\text{вых}}) - \frac{R_2}{R_1} C_2 \frac{dU_{\text{вых}}}{dt}. \quad (2.85)$$

Из (2.81) учитывая (2.84) и (2.85)

$$i_2 = \frac{1}{R_1}(U_{\text{ex}} - U_{\text{блх}}) - \frac{R_2}{R_1}C_2 \frac{dU_{\text{блх}}}{dt} - C_2 \frac{dU_{\text{блх}}}{dt}. \quad (2.86)$$

Дифференцируем по времени (2.86) и подставляем его (2.85), (2.86)

$$\frac{dU_{\text{ex}}}{dt} - \frac{dU_{\text{блх}}}{dt} - R_2C_2 \frac{d^2U_{\text{блх}}}{dt^2} + \frac{1}{C_1} \left[\frac{1}{R_1}(U_{\text{ex}} - U_{\text{блх}}) - \frac{R_2}{R_1}C_2 \frac{dU_{\text{блх}}}{dt} - C_2 \frac{dU_{\text{блх}}}{dt} \right] = \frac{dU_{\text{ex}}}{dt}$$

Сокращаем в левой и правой частях $\frac{dU_{\text{ex}}}{dt}$ и перегруппировываем

$$R_2C_2 \frac{d^2U_{\text{блх}}}{dt^2} + \left[1 + \frac{C_2}{C_1} \left(1 + \frac{R_2}{R_1} \right) \right] \frac{dU_{\text{блх}}}{dt} + \frac{1}{R_1C_1}U_{\text{блх}} = \frac{1}{R_1C_1}U_{\text{ex}}$$

Множим на R_1C_1

$$R_1C_1R_2C_2 \frac{d^2U_{\text{блх}}}{dt^2} + [R_1C_1 + C_2(R_1 + R_2)] \frac{dU_{\text{блх}}}{dt} + U_{\text{блх}} = U_{\text{ex}}. \quad (2.87)$$

Это звено 2-го порядка. Оно как и дифференциальное уравнение (2.78) может быть представлено системой канонических уравнений (коэффициенты которых справедливы для (2.87)).

$$\begin{cases} y_1' = a_{1,1}y_1 + a_{1,2}y_2 + b_{1,1}x_1, & (2.88) \end{cases}$$

$$\begin{cases} y_2' = a_{2,1}y_1 + a_{2,2}y_2 + b_{2,1}x_1, & (2.89) \end{cases}$$

где $y_1 = U_{\text{блх}}$; $y_2 = \frac{dU_{\text{блх}}}{dt}$;

$$a_{1,1} = 0; \quad a_{1,2} = 1; \quad b_{1,1} = 0;$$

$$a_{2,1} = -\frac{1}{R_1C_1R_2C_2}; \quad a_{2,2} = -\frac{R_1C_1 + C_2(R_1 + R_2)}{R_1C_1R_2C_2}; \quad b_{2,1} = \frac{1}{R_1C_1R_2C_2}.$$

3. Компьютерные методы исследования динамики объектов с сосредоточенными параметрами

В пособии [6] были рассмотрены модули Serv, GromS, обеспечивающие простой и удобный интерфейс для формирования интерактивных программ на Турбо Паскале, ориентированных на работу в графическом режиме. В настоящей главе при разработке отдельных программ, фиксирующих рассмотренные алгоритмы, а также в демонстрационно-отладочных программах, листинги которых приводятся в Приложении, авторы пользовались средствами выше упомянутых модулей, рассчитывая, что читатель уже ознакомлен с ними.

3.1. Динамика объектов, задаваемых системой обыкновенных дифференциальных уравнений

Как было показано в предыдущей главе математическая модель объекта с сосредоточенными параметрами может быть представлена в общем случае в виде системы обыкновенных дифференциальных уравнений в канонической форме

$$\frac{dy_z}{dt} = \sum_{s=1}^n A_{z,s} y_s + \sum_{s=1}^m B_{z,s} x_s, \quad 1 \leq z \leq n, \quad (3.1)$$

где y_z - z -ый выходной сигнал; $1 \leq z \leq n$, x_s - s -ый входной сигнал; $A_{z,s}$, $B_{z,s}$ - матрицы коэффициентов. В общем случае отдельные элементы матриц $A_{z,s}$, $B_{z,s}$ могут зависеть от значений входных и выходных параметров объекта, от времени и других факторов, но в пределах данной главы их будем рассматривать как константы, т.е. как числа типа `real`.

Математическая модель может использоваться при исследовании автоматических систем и сама по себе (как таковая), или же она входит в состав более сложной системы, например, технологической схемы, состоящей из ряда агрегатов, каждый из которых описывается моделью типа (3.1) и т.д.

Ниже рассматриваются несколько компьютерных алгоритмов, позволяющих получить из (3.1) передаточные функции отдельных каналов объекта, рассчитать переходные или частотные характеристики, разложить передаточную функцию заданного канала в ряд Маклорена. Последнее может быть полезным для определения аппроксимирующей передаточной функции.

3.1.1. Определение передаточных функций

Определение передаточной функции отдельного канала, передаточных функций определенной группы каналов или даже передаточных функций всех возможных каналов в объекте может представлять собой интерес для исследователя. В общем случае решение такой задачи сводится к применению преобразования Лапласа над системой дифференциальных уравнений (3.1) при нулевых начальных условиях с последующим решением системы n алгебраических уравнений относительно изображения выхода интересующего сигнала или нужной передаточной функции. Продемонстрируем компьютерный вариант этого алгоритма.

Выполним над (3.1) преобразование Лапласа при нулевых начальных условиях.

$$\sum_{s=1}^n C_{z,s}(p)\bar{y}_s = -\sum_{s=1}^m B_{z,s}\bar{x}_s, \quad 1 \leq z \leq n. \quad (3.2)$$

Здесь:

$$C_{z,s}(p) = \begin{cases} A_{z,s} - p & \text{при } s = z, \\ A_{z,s} & \text{при } s \neq z, \end{cases}$$

\bar{x}_s, \bar{y}_s - изображения по Лапласу от x_s и y_s

$$\begin{cases} \bar{x}_s = \int_0^{\infty} x_s e^{-pt} dt, \\ \bar{y}_s = \int_0^{\infty} y_s e^{-pt} dt. \end{cases} \quad (3.3)$$

Если из всех возможных входов x_s ($1 \leq s \leq m$) предоставить ненулевое приращение только входу x_{Ninp} и поделить (3.3) на \bar{x}_{Ninp} , то система (3.3) приводится к виду:

$$\sum_{s=1}^n C_{z,s}(p)W_{Ninp,s}(p) = -B_{z,Ninp}, \quad 1 \leq z \leq n. \quad (3.4)$$

Здесь: $W_{Ninp,s}(p)$ - передаточная функция канала „ $x_{Ninp} \rightarrow y_s$ ”.

Решим теперь систему (3.4) относительно $W_{Ninp,s}(p)$ методом Гаусса.

Расширенная матрица Rm коэффициентов этой системы будет иметь такую структуру:

	1	2	3	...	n	$n + 1$
1	$A_{1,1} - p$	$A_{1,2}$	$A_{1,3}$...	$A_{1,n}$	$-B_{1,Ninp}$
2	$A_{2,1}$	$A_{2,2} - p$	$A_{2,3}$...	$A_{2,n}$	$-B_{2,Ninp}$
Rm 3	$A_{3,1}$	$A_{3,2}$	$A_{3,3} - p$...	$A_{3,n}$	$-B_{3,Ninp}$
...
n	$A_{n,1}$	$A_{n,2}$	$A_{n,3}$...	$A_{n,n} - p$	$-B_{n,Ninp}$
	$W_{Ninp,1}(p)$	$W_{Ninp,2}(p)$	$W_{Ninp,3}(p)$		$W_{Ninp,n}(p)$	

Под каждым из первых n столбцов матрицы показана передаточная функция (неизвестная), коэффициентами при которой являются элементы соответствующего столбца.

Структуру матрицы Rm отнесем к типу:

```
type M2C = array[1..n, 1..(n+1)] of coef,
где type Coef = array[-1..31] of real.
```

Тип Coef в данном случае используется для размещения в нем информации о полиноме: $A(p) = a_k p^k + \dots + a_1 p + a_0$, по схеме:

-1	0	1	2	...	k	...	31
k	a_0	a_1	a_2	...	a_k		

Решение системы (3.4) формируем по схеме Гаусса, то есть путем приведения матрицы Rm к треугольному виду, когда все элементы этого массива ниже главной диагонали превращаются в нули (прямой ход схемы Гаусса).

Обнуление элемента Rm[z, i] выполняем вычитая из элементов z-ой строки, умноженных на Rm[i, i], элементы i-ой строки, умноженные на Rm[z, i]. Результаты поэлементного вычитания заносятся в z-ую строку.

После завершения прямого хода последняя n -ая строка матрицы Rm будет соответствовать уравнению:

$$Rm[n, n] \cdot W_{Ninp, n}(p) = Rm[n, n+1],$$

откуда:

$$W_{Ninp, n}(p) = \frac{Rm[n, n+1]}{Rm[n, n]}. \quad (3.5)$$

Известно, что все передаточные функции объекта с сосредоточенными параметрами, а объект, описываемый системой (3.1), является именно таким, могут быть представлены в виде, когда их знаменатели одинаковые (имеют общий знаменатель). Таким образом $Rm[n, n]$ в формуле (3.5) может быть принят за этот самый общий знаменатель.

Итак, при выполнении обратного хода схемы Гаусса достаточно определить лишь недостающие числители передаточных функций. $W_{Ninp, z}$, где z изменяется в диапазоне от $n-1$ до 1 (числитель для $W_{Ninp, n}$ уже фигурирует в (3.5), то есть в ячейке $Rm[n, n+1]$). Остальные числители будем заносить в ячейки Rm по схеме: „числитель $W_{Ninp, z}(p) \rightarrow$ ячейка $Rm[z, n+1]$ ”.

Легко заметить, что незначительной модификацией матрицы Rm , а именно внесением в $(n+s)$ -ый столбик значений $-B_{z, s}$ ($1 \leq s \leq m$) и при обслуживании в процессе прямого хода столбиков до $(n+m)$ -го включительно, можно обеспечить в процессе выполнения обратного хода получение числителей передаточных функций $W_{s, z}(p)$, $1 \leq s \leq m$, $1 \leq z \leq n$, то есть определение всех возможных в данном объекте передаточных функций. Они будут записаны в ячейках массива Rm по схеме: „числитель $W_{s, z}(p) \rightarrow$ ячейка $Rm[z, n+s]$ ”. Общий же знаменатель, как уже отмечалось, в ячейке $Rm[n, n]$.

Описанный алгоритм реализован в процедуре $WpSU$.

```
procedure WpSU(A, B:M2);
var i, z, s:integer;
    E, F, C, D:coef;
begin
```

```

N:=round(A[1,-1]); M:=round(B[1,-1]);
for z:=1 to n do
  begin
    for s:=1 to n do
      if s=z
        then
          begin
            Rm[z,s,-1]:=1; Rm[z,s,0]:=A[z,z];
            Rm[z,s,1]:=-1
          end
        else
          begin
            Rm[z,s,-1]:=0; Rm[z,s,0]:=A[z,s]
          end;
        for S:=1 to m do
          begin
            Rm[z,n+s,-1]:=0; Rm[z,n+s,0]:=-B[z,s]
          end
        end;
      end;
    for i:=1 to n-1 do
      begin
        C:=Rm[i,i];
        for z:=i+1 to n do
          if not((Rm[z,i,-1]=0) and (Rm[z,i,0]=0)) then
            begin
              D:=Rm[z,i];
              for s:=i+1 to n+m do
                begin
                  UmnPol(Rm[z,s],C,E);
                  UmnPol(Rm[i,s],D,F);
                  AlSumPol(E,F,'-',Rm[z,s]);
                  redPol(1e-9,Rm[z,s])
                end
              end
            end
          end;
        if Rm[n,n,n]<0 then
          for S:=n to n+m do
            for i:=0 to n do Rm[n,s,i]:=-Rm[n,s,i];
            for z:=n-1 downto 1 do
              for s:=1 to m do
                begin
                  c[-1]:=0; c[0]:=0;
                  for i:=z+1 to n do
                    begin
                      UmnPol(Rm[z,i],Rm[i,n+s],D);

```

```
        AlSumPol (C, D, '+', C)
    end;
    UmnPol (Rm [z, n+s], Rm [n, n], D);
    AlSumPol (D, C, '-', C);
    DelPol (C, Rm [z, z], Rm [z, n+s], D)
end
end;
```

3.1.2. Расчет переходной характеристики

Компьютерные методы численного интегрирования системы обыкновенных дифференциальных уравнений (3.1) были рассмотрены в работах [4–6, 9, 12]. Для вычисления переходной характеристики канала „ $X_{Nin} \rightarrow Y_{Nout}$ ” воспользуемся методом Ейлера. Поскольку компьютер разрешает выбирать шаг интегрирования как угодно малым, то низкий (первый) порядок точности этого метода не является преградой для достижения нужной точности интегрирования. Шаг по времени при численном интегрировании с использованием этого метода реализуется в подпрограмме EilSU. Представим листинг этой подпрограммы:

```

procedure EilSU(A,B: Coef2; Mx0:coef; V0:real;
               var Y:coef);
var z,s,n,m: integer;
    sum: real;
begin
  n:=round(A[1,-1]); m:=round(B[1,-1]);
  for z:=1 to n do
    begin
      sum:=0;
      for s:=1 to n do sum:=sum+A[z,s]*Y[s];
      for s:=1 to m do sum:=sum+B[z,s]*Mx0[s];
      Y[z]:=Y[z]+V0*sum
    end
  end;
end;

```

Здесь A, B - массивы коэффициентов системы (3.1); $Mx0$ - массив значений входных сигналов на момент времени $t = \nu\tau$ (начало шага).

Схема массива $Mx0$:

	-1	0	1	2	3	...	m	...	31
$Mx0$	m		$x_{1,\nu}$	$x_{2,\nu}$	$x_{3,\nu}$...	$x_{m,\nu}$...	

Относительно расчета переходной характеристики канала, входом

которого есть $X_{N_{inp}}$, этот массив на все время расчета приобретает вид:

	-1	0	1	2	3	...	N_{inp}	...	m	...	31
Mx0	m		0	0	0	...	1	...	0	...	

$V0$ - величина шага интегрирования по времени.

Y - массив параметров состояния (выходов) системы (3.1).

Структура массива Y на момент времени $t = \nu\tau$

	-1	0	1	2	3	...	n	...	31
Y	n		$y_{1,\nu}$	$y_{2,\nu}$	$y_{3,\nu}$...	$y_{n,\nu}$...	

В начале расчета переходной характеристики массив Y заполняется нулями (ячейки с номерами от 1 к n). Это соответствует нулевым начальным условиям, которые входят в определение переходной характеристики. При каждом очередном вызове процедуры EilSU массив Y обновляется в соответствии с состоянием объекта на конец очередного шага.

Процедура HtSU формирует массив Ht:CoefL ординат переходной характеристики “заказанного” канала, который имеет следующую структуру:

	-1	0	1	2	3	...	L	$L+1$...	601
Ht	L	h_0	h_1	h_2	h_3	...	h_L	Dt	...	

Здесь $h_s = h(t)|_{t=s\tau}$, Dt - шаг квантования по времени между соседними ординатами (τ). Приведем процедуру HtSU

```

Procedure HtSU(A,B:Coef2; L,Ninp,Nout,ks:integer, D:real;
               var Ht:Coef1);
var Dt,V0:real;
    Y,Mx:Coef;
    n,m,z,s:integer;
begin
    n:=round(A[1,-1]); m:=round(B[1,-1]);

```

```

Mx[-1]:=m; Y[-1]:=n;
for s:=0 to m do Mx[s]:=0;
  Mx[Ninp]:=1;
for s:=0 to n do Y[s]:=0;
  Ht[-1]:=L; Ht[0]:=0;
Dt:=D/L; V0:=Dt/ks; Ht[L+1]:=Dt;
for z:=1 to L do
  begin
    for s:=1 to ks do
      EilSU(A,B,Mx,V0,Y);
      Ht[z]:=Y[Nout]
    end
  end
end;

```

Здесь D - время наблюдения переходной характеристики, ks - коэффициент дробления шага по времени.

3.1.3. Расчет частотных характеристик

Конечно, имея передаточную функцию нужного канала, заменяя p на $j\omega$ можно рассчитать частотные характеристики этого канала. Но относительно объектов, которые описываются системой дифференциальных уравнений типа (3.1), расчет частотных характеристик можно выполнить на компьютере непосредственно, минуя стадию определения передаточной функции.

Итак, пусть модель объекта задана системой (3.1). Будем также считать, что рассматривается „верр” каналов, общим входом которых есть X_{Ninp} , а выходами все возможные выходы y_s ($1 \leq s \leq n$). Пусть ненулевое приращение имеет вход x_{Ninp} , приращения других входов будем считать нулевыми.

После преобразования по Лапласу при нулевых начальных условия и деления системы (3.1) на x_{Ninp} , имеем:

$$pW_{Ninp,z}(p) = \sum_{s=1}^n A_{z,s}W_{Ninp,s}(p) + B_{z,Ninp}, \quad 1 \leq z \leq n, \quad (3.6)$$

где: $W_{Ninp,s}(p)$ - передаточная функция по каналу „ $x_{Ninp} \rightarrow y_s$ ”.

Выполняем в (3.6) замену $p = j\omega$

$$j\omega(R_z + iI_z) = \sum_{s=1}^n A_{z,s}(R_s + iI_s) + B_{z,Ninp}, \quad 1 \leq z \leq n, \quad (3.7)$$

где R_z, I_z - вещественно- и мнимо-частотная характеристики канала „ $X_{Ninp} \rightarrow y_z$ ”, R_s, I_s - то же для канала „ $X_{Ninp} \rightarrow y_s$ ”.

Каждое из комплексных уравнений (3.7) разлагается на два независимых уравнения (для вещественной и мнимой частей). Для z -го уравнения имеем:

$$\begin{cases} \sum_{s=1}^n A_{z,s}R_s + \omega I_z = -B_{z,Ninp}, & (3.8) \\ -\omega R_z + \sum_{s=1}^n A_{z,s}I_s = 0. & (3.9) \end{cases}$$

В результате из (3.8), (3.9) получаем систему $2n$ уравнений относительно $2n$ неизвестных $R_s, I_s, 1 \leq s \leq n$. Расположим неизвестные в таком порядке:

$R_1, R_2, \dots, R_n, I_1, I_2, \dots, I_n$, а в системе уравнений сначала поставим уравнения для вещественных частей $1 \leq z \leq n$, а потом уже уравнения для мнимых частей $1 \leq z \leq n$. Тогда расширенная матрица системы будет иметь такую структуру:

	1	2	3	...	n	$n+1$	$n+2$	$n+3$...	$2n$	$2n+1$
1	$A_{1,1}$	$A_{1,2}$	$A_{1,3}$...	$A_{1,n}$	ω	0	0	...	0	$-B_{1,N_{inp}}$
2	$A_{2,1}$	$A_{2,2}$	$A_{2,3}$...	$A_{2,n}$	0	ω	0	...	0	$-B_{2,N_{inp}}$
3	$A_{3,1}$	$A_{3,2}$	$A_{3,3}$...	$A_{3,n}$	0	0	ω	...	0	$-B_{3,N_{inp}}$
...
n	$A_{n,1}$	$A_{n,2}$	$A_{n,3}$...	$A_{n,n}$	0	0	0	...	ω	$-B_{n,N_{inp}}$
$n+1$	$-\omega$	0	0	...	0	$A_{1,1}$	$A_{1,2}$	$A_{1,3}$...	$A_{1,n}$	0
$n+2$	0	$-\omega$	0	...	0	$A_{2,1}$	$A_{2,2}$	$A_{2,3}$...	$A_{2,n}$	0
$n+3$	0	0	$-\omega$...	0	$A_{3,1}$	$A_{3,2}$	$A_{3,3}$...	$A_{3,n}$	0
...
$2n$	0	0	0	...	$-\omega$	$A_{n,1}$	$A_{n,2}$	$A_{n,3}$...	$A_{n,n}$	0
	R_1	R_2	R_3	...	R_n	I_1	I_2	I_3	...	I_n	

Под матрицей Ms показаны неизвестные, коэффициентами при которых является элементы соответствующего столбика. Итак, массив $x:Coef$ значений неизвестных имеет такую структуру:

-1	0	1	2	3	...	n	$n+1$	$n+2$	$n+3$...	$2n$...	31
$2n$		R_1	R_2	R_3	...	R_n	I_1	I_2	I_3	...	I_n	...	

Расчет частотных характеристик канала „ $X_{N_{inp}} \rightarrow y_{N_{out}}$ ” для конкретного значения частоты ω определяем с помощью подпрограммы UrGod.

```

procedure UrGod;
var z,s:integer; Ms:Matr; R:Coef;
begin
  for z:=1 to n do
    begin
      for s:=1 to n do
        begin
          Ms[z,s]:=A[z,s];
          Ms[z+n,s+n]:=A[z,s];
          if s=z
            then

```

```

begin
  Ms[z, s+n] := w;
  Ms[z+n, s] := -w
end
else
begin
  Ms[z, s+n] := 0; Ms[z+n, s] := 0
end
end;
Ms[z, 2*n+1] := -B[z, Ninp];
Ms[z+n, 2*n+1] := 0
end;
SystUr(2*n, Ms, R); x := R[Nout]; Y := R[Nout+n];
end;

```

3.1.4. Расчет коэффициентов ряда Маклорена

Как было показано, система канонических уравнений (3.1) приводится к виду:

$$\sum_{s=1}^n C_{z,s}(p)W_{Ninp,s}(p) = -B_{z,Ninp}, \quad 1 \leq z \leq n, \quad (3.10)$$

где:

$$C_{z,s}(p) = \begin{cases} A_{z,s} - p, & \text{при } s = z \\ A_{z,s}, & \text{при } s \neq z \end{cases}$$

Здесь N_{inp} - номер входного сигнала, общего для всех каналов, которые описываются передаточными функциями $W_{Ninp,s}(p)$, $1 \leq s \leq n$.

Будем разлагать $W_{Ninp,s}(p)$ в ряд Маклорена:

$$W_{Ninp,s}(p) = \sum_{r=0}^{\infty} e_{r,s} p^r, \quad (3.11)$$

где:

$$e_{r,s} = \frac{1}{s!} \left. \frac{d^s W_{Ninp,s}(p)}{dp^s} \right|_{p=0}. \quad (3.12)$$

Здесь r - номер коэффициента в ряде.

При $p = 0$ система (3.10) приобретает вид:

$$\sum_{s=1}^n A_{z,s} e_{0,s} = -B_{z,Ninp}, \quad 1 \leq z \leq n. \quad (3.13)$$

Система (3.13) – это система линейных алгебраических уравнений относительно $e_{0,s}$, $1 \leq s \leq n$, и она может быть решена, например, по схеме Гаусса (процедура `SystUr`, которой передается соответственно сформированная расширенная матрица).

А теперь продифференцируем (3.10) относительно p :

$$\sum_{s=1}^n C_{z,s}(p) \frac{dW_{Ninp,s}(p)}{dp} = W_{Ninp,z}(p), \quad 1 \leq z \leq n. \quad (3.14)$$

При $p = 0$ с учетом (3.12) получаем:

$$\sum_{s=1}^n A_{z,s} e_{1,s} = e_{0,z}, \quad 1 \leq z \leq n. \quad (3.15)$$

Если считать, что $e_{0,z}$ известны как решение системы (3.13), то система (3.15) может быть решена относительно $e_{1,s}$, $1 \leq s \leq n$.

Дифференцируем (3.14) относительно p :

$$\sum_{s=1}^n C_{z,s}(p) \frac{d^2 W_{Ninp,s}(p)}{dp^2} = 2 \frac{dW_{Ninp,z}(p)}{dp}, \quad 1 \leq z \leq n. \quad (3.16)$$

При $p = 0$ с учетом (3.12) при $s = 2$ будем иметь:

$$\sum_{s=1}^n A_{z,s} e_{2,s} = e_{1,z}, \quad 1 \leq z \leq n. \quad (3.17)$$

Когда $e_{1,z}$, $1 \leq z \leq n$ известны как решение системы (3.15), то система (3.17) может быть решена относительно $e_{2,s}$, $1 \leq s \leq n$.

Дифференцируем (3.16) относительно p :

$$\sum_{s=1}^n C_{z,s}(p) \frac{d^3 W_{Ninp,s}(p)}{dp^3} = 3 \frac{d^2 W_{Ninp,z}(p)}{dp^2}, \quad 1 \leq z \leq n. \quad (3.18)$$

При $p = 0$ с учетом (3.12) при $S = 3$:

$$\sum_{s=1}^n A_{z,s} e_{3,s} = \frac{e_{2,z}}{2!}, \quad 1 \leq z \leq n. \quad (3.19)$$

Система (3.19) решается относительно $e_{3,s}$, $1 \leq s \leq n$.

Дифференцируем (3.18) относительно p :

$$\sum_{s=1}^n C_{z,s}(p) \frac{d^4 W_{Ninp,s}(p)}{dp^4} = 4 \frac{d^3 W_{Ninp,z}(p)}{dp^3}, \quad 1 \leq z \leq n, \quad (3.20)$$

при $p = 0$:

$$\sum_{s=1}^n A_{z,s} e_{4,s} = \frac{e_{3,z}}{3!}, \quad 1 \leq z \leq n. \quad (3.21)$$

и т.д.

В общем случае имеем рекуррентную систему уравнений:

$$\sum_{s=1}^n A_{z,s} e_{r,s} = \frac{e_{r-1,z}}{(r-1)!}, 1 \leq z \leq n. \quad (3.22)$$

которая последовательно решается при $r=1, 2, \dots, Ne \dots$

Начальная же (при $r = 0$) система уравнений имеет вид системы (3.13).

Этот алгоритм формирования массива E:Coef коэффициентов ряда (3.11) фиксируем в подпрограмме ESU.

```

Procedure ESU (A,B:Coef2; Ninp, Nout, Ne: integer;
               var E:Coef);
var Z,n,r: integer;
    Zn: real;
    C:Coef2;
    Mx: Coef;
Begin
  E [-1] := Ne; n:=round(A[1,-1]); C:=A;
  for Z:=1 to n do C[Z,n+1]:=-B[Z, Ninp] ;
  Systur(n,C,Mx); E[0]:=Mx[Nout].;
  Zn:=1;
  for r:=1 to Ne do
    Begin
      C:=A; Zn:=Zn*r;
      for z:=1 to n do
        C[z,n+1]:=E[r-1]/Zn;
      SystUr(n,C,Mx); E[r]:=Mx[Nout]
    end
  end;
end;

```

3.2. Характеристики канала, задаваемого дробно-рациональной передаточной функцией

Если речь идет об отдельном канале (или единственном), то одной из наиболее привычных, удобных и компактных форм задания динамических свойств этого канала является его передаточная функция. Для объектов с сосредоточенными параметрами, как уже неоднократно упоминалось, передаточная функция является дробно-рациональной (равна отношению двух полиномов). Часто для расширения функциональных возможностей (в первую очередь для отображения эффектов транспортных и емкостных запаздываний) в подобную передаточную функцию вводят множитель $e^{-p\tau}$. Это, собственно говоря, эффект, характерный для объектов с распределенными параметрами. Будем его рассматривать как своеобразный мостик, связывающий объекты с сосредоточенными и распределенными параметрами. При рассмотрении переходных и частотных характеристик отдельного канала наличие множителя $e^{-p\tau}$ в его передаточной функции почти не усложняет расчета, а при исследовании замкнутых систем рядом методов даже упрощает анализ.

3.2.1. Расчет переходной характеристики

В [6] было рассмотрено формирование системы замещения для дробно-рациональной передаточной функции, там же приведен листинг процедуры SystZam.

Шаг по времени при численном интегрировании системы замещения, например, методом Ейлера реализует подпрограмма EilWp. Массив ординат Ht:Coef переходной характеристики, которая соответствует передаточной функции $W(p) = B(p)/A(p)$, формируется подпрограммой Htwp.

```
Procedure Htwp(B, A, Coef; L, ks:integer; D: real;
              var Ht: CoefL);
var Dt, v0 : real;
    Y, K: Coef;
    z, s: integer;
begin
  SystZam(A, B, K);
  for S:=0 to round(A[-1]) do Y[s]:=0;
  Dt:=D/L; v0:=Dt/ks;
  Ht[-1]:=L; Ht[L+1]:=Dt; Ht[0]:=0;
  for z:=1 to L do
  begin
    for s:=1 to ks do
      EilWp(A, K, 1, 1, v0, y);
      Ht[z]:=Y[0]
    end
  end;
end;
```

На аналитическом способе расчета $h(t)$ в рассмотренном объекте мы не останавливаемся, поскольку он намного более трудоемок сравнительно с приведенным и кроме того, как показывает опыт в ряде случаев, например, когда полином $A(p)$ имеет несколько рядом расположенных корней (кратных или почти кратных) аналитический метод может приводить к относительно большим погрешностям при вычислении коэффициентов аналитических формул, что обуславливается неизбежными погрешностями в определении корней полинома $A(p)$. Корни $A(p)$ определяют структуру решения, а погрешности в определении типа корней (кратные, простые,

вещественные, комплексные) приводят к тому, что может быть выбрана неадекватная структура решения. Анализ подобной ситуации приводится в [4].

3.2.2. Расчет частотных характеристик

Для обобщения рассмотрим случай когда передаточная функция имеет вид:

$$W(p) = \frac{B(p)}{A(p)} e^{-p\tau}. \quad (3.23)$$

Заменяем переменную $p = j\omega$ и обозначаем:

$$B(j\omega) = R_1 + jI_1,$$

$$A(j\omega) = R_2 + jI_2,$$

$$e^{-j\omega\tau} = R_3 + jI_3,$$

тогда:

$$W(j\omega) = \frac{R_1 + jI_1}{R_2 + jI_2} (R_3 + jI_3),$$

где $R_3 = \cos(\omega\tau)$; $I_3 = -\sin(\omega\tau)$.

Преобразуем сначала дробь:

$$\frac{R_1 + jI_1}{R_2 + jI_2} = \frac{(R_1 + jI_1)(R_2 - jI_2)}{R_2^2 + I_2^2} = R_4 + jI_4,$$

где:

$$R_4 = \frac{R_1R_2 + I_1I_2}{R_2^2 + I_2^2},$$

$$I_4 = \frac{I_1R_2 - I_2R_1}{R_2^2 + I_2^2},$$

после чего:

$$W(j\omega) = (R_4 + jI_4)(R_3 + jI_3) = x + jy,$$

где:

$$x = R_4R_3 - I_4I_3,$$

$$y = I_4R_3 + I_3R_4.$$

Рассмотренный алгоритм реализует подпрограмма UrGod.

```
Procedure UrGod;  
var R1, I1, R2, I2, R3, I3, R4, I4, Zn: real;
```

```

begin
  HorComp (B, 0, W, R1, I1);
  HorComp (A, 0, W, R2, I2);
  Zn:=sqr (R2) +sqr (I2);
  R4:=(R1*R2+I1+I2);
  I4:=(I1*R2-I2*R1) /Zn;
  Zn:=W* Tau;
  R3:=cos (Zn);
  I3:=-sin (Zn);
  X:=R4*R3-I4*I3;
  Y:=I4*R3+I3*R4
end;

```

Здесь B , A - массивы типа `Coef` коэффициентов полиномов $B(p)$ и $A(p)$.

Tau - запаздывание τ , W частота ω .

3.2.3. Вычисление коэффициентов ряда Маклорена

Пусть передаточная функция задана отрезком ряда Маклорена

$$W(p) = \sum_{s=0}^n e_s p^s.$$

Тогда соответствующая ей амплитудно-фазовая характеристика будет

$$W(j\omega) = \sum_{s=0}^n e_s (j\omega)^s = \operatorname{Re}(\omega) + j \operatorname{Im}(\omega),$$

где

$$\operatorname{Re}(\omega) = \sum_{s=0}^{n \operatorname{div} 2} (-1)^s e_{2s} \omega^{2s},$$

$$\operatorname{Im}(\omega) = \begin{cases} 0 & \text{при } n = 0, \\ \omega \sum_{s=0}^{(n-1) \operatorname{div} 2} (-1)^s e_{2s+1} \omega^{2s} & \text{при } n \geq 1. \end{cases}$$

Процедура UrGod для данного случая приобретает вид

```

Procedure UrGod;
var Zn,s:integer; w2,w2s:real;
begin
  x:=0; w2:=sqr(w); w2s:=1; Zn:=1;
  for s:=0 to n div 2 do
  begin
    x:=x+E[2*s]*w2s*Zn;
    Zn:=-Zn; w2s:=w2s*w2
  end;
  y:=0; w2s:=1; Zn:=1;
  if n>0 then
    for s:=0 to (n-1) div 2 do
      begin
        y:=y+E[2*s+1]*w2s*Zn;
        w2s:=w2s*w2; Zn:=-Zn
      end;
    y:=w*y
  end;
или
Procedure UrGod;
begin
  HorComp(E, 0, w, X, Y)
end;

```

3.3. Характеристики канала, задаваемого массивом ординат импульсной характеристики

Импульсную характеристику можно получить аналитическим путем как оригинал от передаточной функции. Это вариант хорош, если передаточная функция имеется в наличии и она не слишком сложна. Экспериментально импульсную характеристику удобнее получить не непосредственно (технически затруднительно достаточно точно реализовать единичный импульс в качестве входного сигнала), а через переходную. Единичный ступенчатый входной сигнал технически реализовать намного проще, чем единичный импульс. Импульсная характеристика при наличии переходной получается элементарным численным дифференцированием последней.

В данном параграфе будем считать, что импульсная характеристика задается массивом `Mg:Coef1` ее ординат. Предполагается, что время наблюдения D выбрано таким, что при $t > D$ импульсная характеристика практически равна нулю. Считается, что характер импульсной характеристики таков, при котором допустимо линейное интерполирование между ее соседними ординатами.

3.3.1. Расчет переходной характеристики

Переходная и импульсная характеристики, как известно, связаны между собою соотношениями [2]:

$$g(t) = \frac{d}{dt}[h(t)], \quad (3.24)$$

$$h(t) = \int_0^t g(t) dt. \quad (3.25)$$

Если имеем в наличии массив Ht или Gt, типа CoefL, такой, что допускает линейную интерполяцию, то пересчет одного массива в другой можно реализовать одной из подпрограмм HtGt или GtHt, таких как:

```
Procedure HtGt (Ht: CoefL; var Gt: CoefL);
var s, L: integer;
    Dt: real;
begin
    L:=round(Ht[-1]); Dt:= Ht[L+1];
    Gt[-1]:=L; Gt[L+1]:=Dt;
    for s:=1 to L do
        Gt[s]:=(Ht[s] - Ht[s-1]) /Dt
    end;
```

```
Procedure GtHt (Gt: CoefL; var Ht:CoefL);
var s, L: integer;
    Dt: real;
begin
    L:=round(Gt[-1]); Dt:=Gt[L+1];
    Ht[-1]:=L; Ht[L+1]:=Dt; Ht[0]:=0;
    for s:=1 to L do
        Ht[s]:=Ht[s-1]+(Gt[s-1]+Gt[s])/2*Dt
    end;
```

В дальнейшем, учитывая такой простой алгоритм пересчета $h(t)$ в $g(t)$ и обратно, мы будем ограничиваться расчетом одной из характеристик $h(t)$ или $g(t)$, считая, что другая может быть легко рассчитана на базе имеющейся.

3.3.2. Расчет частотных характеристик

Известно, что:

$$W(p) = L[g(t)]$$

или

$$W(p) = \int_0^{\infty} g(t)e^{-pt} dt.$$

Заменяем p на $j\omega$:

$$\operatorname{Re}(\omega) + j \operatorname{Im}(\omega) = \int_0^{\infty} g(t)[\cos(\omega t) - j \sin(\omega t)] dt.$$

откуда:

$$\operatorname{Re}(\omega) = \int_{-\infty}^{\infty} g(t) \cos(\omega t) dt, \quad (3.26)$$

$$\operatorname{Im}(\omega) = -\int_0^{\infty} g(t) \sin(\omega t) dt. \quad (3.27)$$

Если $|g(t)|_{t>D} \leq \varepsilon$, то верхнюю границу интегрирования в (3.26), (3.27)

можно заменить на D :

$$\operatorname{Re}(\omega) = \int_0^D g(t) \cos(\omega t) dt, \quad (3.28)$$

$$\operatorname{Im}(\omega) = -\int_0^D g(t) \sin(\omega t) dt. \quad (3.29)$$

Будем считать, что $g(t)$ задана массивом $Gt : \operatorname{Coef}L$ ее ординат, шаг по времени между соседними ординатами $Dt = D/L$. Если интегрирование в формулах (3.28), (3.29) выполнять по формуле трапеций, считая, что подынтегральные функции при $t=0$ и при $t=D$ практически равняются 0, то:

$$\operatorname{Re}(\omega) = Dt \sum_{s=1}^{L-1} Gt[s] \cdot \cos(\omega \cdot s \cdot Dt), \quad (3.30)$$

$$\operatorname{Im}(\omega) = -Dt \sum_{s=1}^{L-1} Gt[s] \cdot \sin(\omega \cdot s \cdot Dt). \quad (3.31)$$

Формулы (3.30), (3.31) обеспечивают приемлемую точность при условии, что:

$$\frac{2\pi}{\omega} \geq 20 \cdot Dt,$$

или, что одно и то же:

$$\omega \leq \frac{\pi}{10 \cdot Dt}. \quad (3.32)$$

Если же ω больше, чем это разрешается (3.32), то принимаем, что:

$$Dtw = \frac{Pi}{10 \cdot \omega}, \quad (3.33)$$

откуда:

$$Lw = \text{round}\left(\frac{D}{Dtw}\right), \quad (3.34)$$

а уточненное значение:

$$Dtw = D/Lw, \quad (3.35)$$

и тогда формулы (3.30), (3.31) заменяется на:

$$\text{Re}(\omega) = Dtw \sum_{s=1}^{Lw-1} g(s \cdot Dtw) \cos(\omega \cdot s \cdot Dtw), \quad (3.36)$$

$$\text{Im}(\omega) = -Dtw \sum_{s=1}^{Lw-1} g(s \cdot Dtw) \sin(\omega \cdot s \cdot Dtw). \quad (3.37)$$

Здесь, значения $g(s \cdot Dtw)$ вычисляются с использованием, например, интерполяционного B -сплайна или, что проще, путем линейной интерполяции над элементами массива Gt , которую реализует подпрограмма-функция `IntGt`.

```
Function IntGt(Gt:Coef; t:real:real;
              var L,S:integer; Dt,r:real;)
begin
  L:=round(Gt[-1]); Dt:=Gt[L+1];
  R:=t/dt; s:=ttrunk(r); t:=frac(r);
  IntGt:=Gt[s]+t*(Gt[s+1]-Gt[s])
end;
```


С учетом сказанного вычисление $x = \text{Re}(\omega)$ и $y = \text{Im}(\omega)$ для фиксированного значения частоты ω , то есть W , по массиву $Gt:CoefL$ можно представить в виде процедуры $UrGod$.

```

Procedure UrGod;
var wt, Dtw, g:real;
    S,Lw ;integer;
begin
  X:=0; y:=0;
  If w<pi/(10*Dt)
  then
    begin
      for s:=1 to L-1 do
        begin
          Wt:=w*s*Dt; g:=gt[s];
          X:=x+g*cos (wt);
          Y:=x-g*sin (wt);
        end
      else
        begin
          Dtw:=pi/(10*w);Lw:=round(D/Dtw);
          Dtw:=D/Lw;
          for s:=1 to Lw-1 do
            begin
              Wt:=w*s*Dtw;
              G:=IntGt (Gt, s*Dtw);
              X:=x+g*cos (wt);
              Y:=y-g*sin (wt)
            end;
          X:=x*Dtw; y:=y*Dtw
        end
    end;
end;

```

В последнем варианте процедуры $UrGod$ массив $Gt:CoefL$ ординат импульсной характеристики считается глобальным. При необходимости сформировать массивы $MRe, MIm:CoefL$ ординат вещественно- и мнимочастотных характеристик можно воспользоваться процедурой $ReImWr$. Конечно, процедура $UrGod$ должна в таком случае иметь „начинку”, приведенную в данном параграфе (на базе массива Gt).

3.3.3. Вычисление коэффициентов ряда Маклорена

Передаточная функция $W(p)$, как известно, является изображением по Лапласу от импульсной характеристики $g(t)$, то есть:

$$W(p) = \int_0^{\infty} g(t) e^{-pt} dt. \quad (3.38)$$

Разложим e^{-pt} в степенной ряд (ряд Маклорена):

$$e^{-pt} = 1 - tp + \frac{t^2}{2!} p^2 - \frac{t^3}{3!} p^3 + \dots$$

тогда:

$$W(p) = \int_0^{\infty} g(t) dt - p \int_0^{\infty} g(t) t dt + \frac{p^2}{2!} \int_0^{\infty} g(t) t^2 dt - \frac{p^3}{3!} \int_0^{\infty} g(t) t^3 dt + \dots \quad (3.39)$$

Интеграл типа $\int_0^{\infty} g(t) t^s dt$ называют моментом s -го порядка от импульсной характеристики:

$$M_s = \int_0^{\infty} g(t) t^s dt .$$

Формула (3.39) может тогда быть представлена как:

$$W(p) = \sum_{s=0}^{\infty} (-1)^s \frac{M_s}{s!} p^s, \quad (3.40)$$

или:

$$W(p) = \sum_0^{\infty} E_s p^s, \quad (3.41)$$

где:

$$E_s = \frac{(-1)^s}{s!} M_s = \frac{(-1)^s}{s!} \int_0^{\infty} t^s g(t) dt. \quad (3.42)$$

Формула (3.42) положена в основу подпрограммы Egt.

```
Procedure Egt(Gt:CoefL; Ne:integer; var Kob:real;
              var E:Coef) ;
var L, Z, S, Zn: integer;
    Sum, Dt: real;
```

```

begin
  L:=round (Gt [-1] ) ; Dt:=Gt [L+1]; Kob: =0;
  for S:=0 to L-1 do Kob:=Kob+Gt[S];
  Kob:=Kob*Dt;
  for S:=0 to L do
    Gt[S]:=Gt [S]*S*Dt/Kob;
    Zn:=-1; E[-1]:=Ne; E[0]:=1;
    for S:=1 to Ne do
      begin
        Sum:=0;
        for Z:=1 to L-1 do
          begin
            Sum:=Sum+Gt[z];
            Gt[z]:=Gt[z]*z*Dt/S;
          end;
        E[S]:=Sum*Dt*Zn; Zn:=-Zn
      end
    end;
end;

```

3.4. Динамика канала по его частотным характеристикам

Экспериментальное определение частотных характеристик, как правило, намного более трудоемко по сравнению с определением переходной характеристики. Поэтому в то время как переходные и импульсные задаются в виде массивов типа `CoefL` т.е. достаточно подробно, частотные характеристики чаще известны только для ограниченного количества точек (частот). Но в отдельных областях техники (в радиотехнике, например) частотные характеристики определяются без проблем, поэтому не исключается вариант когда, скажем, вещественно-частотная и (или) мнимо-частотная характеристика задается массивом типа `CoefL`. Да и наличие компьютера создает возможности формирования подобных массивов.

3.4.1. Расчет переходной и импульсной характеристик

Учитывая, что раньше упоминалось о легком перерасчете $h(t)$ в $g(t)$ и наоборот, будем решать поставленную задачу относительно $g(t)$.

Как известно

$$L[g(t)] = W(p),$$

и

$$g(t) = L^{-1}[W(p)]$$

или, в соответствии с формулой для обратного преобразования Лапласа при условии, что $W(p)$ устойчива:

$$g(t) = \frac{1}{2\pi j} \int_{-j\infty}^{j\infty} W(p) e^{pt} dp. \quad (3.43)$$

Вводим замену переменной $p = j\omega$, тогда

$$g(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} W(j\omega) e^{j\omega t} d\omega \quad (3.44)$$

или

$$g(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} [\operatorname{Re}(\omega) + j \operatorname{Im}(\omega)] [\cos(\omega t) + j \sin(\omega t)] d\omega.$$

Раскрываем скобки

$$g(t) = \frac{1}{2\pi} \left\{ \int_{-\infty}^{\infty} \operatorname{Re}(\omega) \cos(\omega t) d\omega - \int_{-\infty}^{\infty} \operatorname{Im}(\omega) \sin(\omega t) d\omega \right\} + \\ + j \frac{1}{2\pi} \left\{ \int_{-\infty}^{\infty} \operatorname{Re}(\omega) \sin(\omega t) d\omega - \int_{-\infty}^{\infty} \operatorname{Im}(\omega) \cos(\omega t) d\omega \right\}.$$

В первых двух интегралах подынтегральные функции четные, а в третьем и четвертом – нечетные. Если функции четные, то пределы интегрирования в них можно изменить таким образом:

$$g(t) = \frac{1}{\pi} \left\{ \int_0^{\infty} \operatorname{Re}(\omega) \cos(\omega t) d\omega - \int_0^{\infty} \operatorname{Im}(\omega) \sin(\omega t) d\omega \right\}. \quad (3.45)$$

Учитывая, что $g(t)|_{t<0} = 0$ (в соответствии с определением), и заменяя в (3.45) t на $-t$ приходим к выводу что:

$$0 = \frac{1}{\pi} \left\{ \int_0^{\infty} \operatorname{Re}(\omega) \cos(\omega t) d\omega + \int_0^{\infty} \operatorname{Im}(\omega) \sin(\omega t) d\omega \right\}.$$

откуда:

$$\int_0^{\infty} \operatorname{Im}(\omega) \sin(\omega t) d\omega = - \int_0^{\infty} \operatorname{Re}(\omega) \cos(\omega t) d\omega.$$

Итак, (3.45) можно представить так:

$$g(t) = \frac{2}{\pi} \int_0^{\infty} \operatorname{Re}(\omega) \cos(\omega t) d\omega \quad (3.46)$$

или:

$$g(t) = - \frac{2}{\pi} \int_0^{\infty} \operatorname{Im}(\omega) \sin(\omega t) d\omega. \quad (3.47)$$

Если исследуемая система имеет частоту среза ω_{zr} такую, что:

$$\begin{cases} \operatorname{Re}(\omega) \Big|_{\omega > \omega_{zr}} \leq \varepsilon, \\ \operatorname{Im}(\omega) \Big|_{\omega > \omega_{zr}} \leq \varepsilon, \end{cases}$$

то верхнюю границу интегрирования в (3.46), (3.47) можно заменить на ω_{zr} :

$$g(t) = \frac{2}{\pi} \int_0^{\omega_{zr}} \operatorname{Re}(\omega) \cos(\omega t) d\omega, \quad (3.48)$$

$$g(t) = - \frac{2}{\pi} \int_0^{\omega_{zr}} \operatorname{Im}(\omega) \sin(\omega t) d\omega. \quad (3.49)$$

Теперь для конкретности будем считать, что $\operatorname{Re}(\omega)$ или $\operatorname{Im}(\omega)$ заданы массивами своих ординат типа `CoefL`, то есть довольно детально с шагом $Dw = \omega_{zr} / L$, при котором допустима линейная интерполяция. Тогда реализация формулы, например, (3.48) может быть оформлена в виде подпрограммы `GtRe`.

```

Procedure GtRe(Re: CoefL; D:real; var Gt:Coef);
var sum, Dt, Dw:real;
    L,Z,S :integer;
begin

```

```

L:=round(Re[-1]); Dw:=Re[L+1];
Dt:=D/L; Gt[-1]:=L; Gt[L+1]:=Dt;
for Z:=1 to L do
  begin
    Sum:=0;
    for s:=0 to L-1 do
      Sum:=Sum+Re[s]*cos(s*Dw*Z*Dt);
    Gt[z]:=2/pi*Sum
  end
end;

```

Аналогично:

```

Procedure GtIm(Im:Coef; D:real; var Gt:Coef);
var Sum,Dt,Dw:real;
    L,Z,S:integer;
begin
  L:=round(Im[-1]); Dw:=Im[L+1];
  Dt:=-D/L; Gt[-1]:=L; Gt[0]:=0; Gt[L+1]:=Dt;
  for Z:=1 to L do
    begin
      Sum:=0;
      for S:=0 to L-1 do
        Sum:=Sum+Im[s]*sin/s*Dw*z*Dt;
      Gt[z]:=-Sum*2/pi
    end
  end;
end;

```

3.4.2. Вычисление коэффициентов ряда Маклорена

Пусть имеем ряд точек амплитудно–фазовой характеристики

$$W(j\omega_s) = R_s + jI_s, \quad 0 \leq s \leq m, \quad (3.50)$$

для частот ω_s .

Значения частот ω_s , вещественно–частотных R_s и мнимо–частотных характеристик I_s занесены в одноименные массивы $\bar{W}, R, I : \text{Coef}$ по схеме

$$W[-1] = n; \quad W[S] = \omega_s, \quad R[S] = R_s, \quad I[S] = I_s.$$

Аппроксимирующую передаточную функцию будем искать в виде

$$W(p) = \sum_{s=0}^n e_s p^s, \quad (3.51)$$

итак, соответствующая ей амплитудно–фазовая характеристика будет

$$W(j\omega) = (e_0 - e_2\omega^2 + e_4\omega^4 - e_6\omega^6) + j(e_1\omega - e_3\omega^3 + e_5\omega^5 - e_7\omega^7). \quad (3.52)$$

Здесь для конкретности принято $n=7$. Результат, разумеется, будет обобщен на произвольное n .

За критерий качества аппроксимации примем сумму квадратов расстояний между соответствующими базовыми и аппроксимирующими точками амплитудно–фазовых характеристик

$$E = \sum_{s=0}^m \left\{ (e_0 - e_2\omega_s^2 - e_4\omega_s^4 - e_6\omega_s^6 - R_s)^2 + (e_1\omega_s - e_3\omega_s^3 + e_5\omega_s^5 - e_7\omega_s^7 - I_s)^2 \right\} \rightarrow \min. \quad (3.53)$$

Для минимизации E необходимо чтобы все ее производные относительно $e_0, e_1, e_2, \dots, e_n$ равнялись нулю.

$$\left. \begin{aligned}
\frac{\partial E}{\partial e_0} &= 2 \sum_{s=0}^m (e_0 - e_2 \omega_s^2 + e_4 \omega_s^4 - e_6 \omega_s^6 - R_s) \omega_s^0 = 0 \\
\frac{\partial E}{\partial e_1} &= 2 \sum_{s=0}^m (e_1 \omega_s - e_3 \omega_s^3 + e_5 \omega_s^5 - e_7 \omega_s^7 - I_s) \omega_s^1 = 0 \\
\frac{\partial E}{\partial e_2} &= 2 \sum_{s=0}^m (e_0 - e_2 \omega_s^2 + e_4 \omega_s^4 - e_6 \omega_s^6 - R_s) (-\omega_s^2) = 0 \\
\frac{\partial E}{\partial e_3} &= 2 \sum_{s=0}^m (e_1 \omega_s - e_3 \omega_s^3 + e_5 \omega_s^5 - e_7 \omega_s^7 - I_s) (-\omega_s^3) = 0 \\
\frac{\partial E}{\partial e_4} &= 2 \sum_{s=0}^m (e_0 - e_2 \omega_s^2 + e_4 \omega_s^4 - e_6 \omega_s^6 - R_s) \omega_s^4 = 0 \\
\frac{\partial E}{\partial e_5} &= 2 \sum_{s=0}^m (e_1 \omega_s - e_3 \omega_s^3 + e_5 \omega_s^5 - e_7 \omega_s^7 - I_s) \omega_s^5 = 0 \\
\frac{\partial E}{\partial e_6} &= 2 \sum_{s=0}^m (e_0 - e_2 \omega_s^2 + e_4 \omega_s^4 - e_6 \omega_s^6 - R_s) (-\omega_s^6) = 0 \\
\frac{\partial E}{\partial e_7} &= 2 \sum_{s=0}^m (e_1 \omega_s - e_3 \omega_s^3 + e_5 \omega_s^5 - e_7 \omega_s^7 - I_s) (-\omega_s^7) = 0
\end{aligned} \right\} (3.54)$$

После сокращения на 2 получаем систему линейных алгебраических уравнений относительно искомых коэффициентов ряда Маклорена. Ее расширенная матрица R имеет вид

	1	2	3	4	5	6	7	8	9
1	$\sum \omega_s^0$	0	$-\sum \omega_s^2$	0	$\sum \omega_s^4$	0	$-\sum \omega_s^6$	0	$\sum R_s \omega_s^0$
2	0	$\sum \omega_s^2$	0	$-\sum \omega_s^4$	0	$\sum \omega_s^6$	0	$-\sum \omega_s^8$	$\sum I_s \omega_s^1$
3	$-\sum \omega_s^2$	0	$\sum \omega_s^4$	0	$-\sum \omega_s^6$	0	$\sum \omega_s^8$	0	$-\sum R_s \omega_s^2$
4	0	$-\sum \omega_s^4$	0	$\sum \omega_s^6$	0	$-\sum \omega_s^8$	0	$\sum \omega_s^{10}$	$-\sum I_s \omega_s^3$
5	$\sum \omega_s^4$	0	$-\sum \omega_s^6$	0	$\sum \omega_s^8$	0	$-\sum \omega_s^{10}$	0	$\sum R_s \omega_s^4$
6	0	$\sum \omega_s^6$	0	$-\sum \omega_s^8$	0	$\sum \omega_s^{10}$	0	$-\sum \omega_s^{12}$	$\sum I_s \omega_s^5$
7	$-\sum \omega_s^6$	0	$\sum \omega_s^8$	0	$-\sum \omega_s^{10}$	0	$\sum \omega_s^{12}$	0	$-\sum R_s \omega_s^6$
8	0	$-\sum \omega_s^8$	0	$\sum \omega_s^{10}$	0	$-\sum \omega_s^{12}$	0	$\sum \omega_s^{14}$	$-\sum I_s \omega_s^7$
	e_0	e_1	e_2	e_3	e_4	e_5	e_6	e_7	Правые части

Под матрицей показаны неизвестные, коэффициентами при которых являются элементы соответствующего столбика матрицы. Формирование расширенной матрицы системы уравнений, ее решение и формирование массива коэффициентов ряда (3.54) описывает подпрограмма AFCE. Ряд (3.54) – это полином n -ой степени и информация о нем в массиве E:Coef размещена именно как для полинома.

```

Procedure AFCE(W,R,I:Coef; n:integer; var E:Coef);
Var Z,S:integer;
    Wk,W2:Coef;
    V:real; R:Matr;
Begin
  m:=round(W[-1]); R[1,1]:=m+1;
  for S:=0 to m do
    begin Wk[S]:=1; W2[S]:=Sqr(W[S]) end;
    for Z:=1 to n+1 do
      begin
        V:=0;
        for S:=0 to m do
          begin
            if odd(Z) then V:=V+R[S]*Wk[S]
              else V:=V+I[S]*Wk[S];
            Wk[S]:=Wk[S]*W[S]
          end;
        If odd((z-1)div 2) then R[Z,n+2]:=-V
          else R[Z,n+2]:=V
        end;
        for S:=0 to m do Wk[S]:=1;
        for Z:=1 to n do
          begin
            V:=0;
            for S:=0 to m do
              begin
                Wk[S]:=Wk[S]*W2[S]; V:=V+Wk[S]
              end;
            for S:=1 to 2*Z-1 do
              if (2*Z+1-S<=n+1) and (S<=n+1) then
                if odd(Z) then R[2*Z+1-S,S]:=-V
                  else R[2*Z+1-S,S]:=V
              end;
            end;
          end;
        Systur(n+1,R,E); E[-1]:=n;

```

```
for S:=0 to n do E[S]:=E[S+1]
end;
```

Приложения

```
Program DemObWp;
      uses   Crt, Graph, Serv, Polynom, Gromk;

const
  D:real=20;
  Ks:integer=1;
  n:integer=2;           Tc:char='H';{alter'G'}
  Kw:real=1.5;
  Wn:real=0.01;
  Wk:real=2;           Tau:real=0;
  Nsc:integer=50;     Eps:real=1e-6;
  Mk:real=1.4;       q:integer=0;
  R:integer=1;

var   A,B,Ksz,My:coef;
      Mz,Ht,Gt:coefl;
      Dt,V0,W,DW:real;
      T20:string[20];

procedure InpPol(Id:string; var A:coef);
  var S,N:integer;
  begin
    Oui('N-stepen polynoma '+Id,N);  A[-1]:=N;
    for S:=N downto 0 do
      begin
        str(s,T10);  Our(Id+'['+T10+']',A[s])
      end
    end;
end;

procedure OutPol ( x:integer; Id:string; A:coef);
  var S,N:integer;
  begin
    N:=round(A[-1]);
    for S:=0 to N do
      begin
        Str(S,T10);  Str(A[s],Ts);
        Outtextxy(X,(S+2)*10,id+'['+T10+']='+Ts)
      end
    end;
end;

procedure SystZam (B,A:coef; var K:coef);
  var S,n,R,m:integer;
```

```

        Sum:=real;
begin
  N:=round(A[-1]); m:=round(B[-1]);
  if m<n then
    for S:=m+1 to n do B[S]:=0;
  K[0]:=B[n]/A[n];
  for s:=1 to n do
    begin
      Sum:=0;
      for R:=0 to S-1 do
        Sum:=Sum+K[R]*A[n-S+R];
      K[S]:=(B[n-s]-Sum)/A[n]
    end
  end;
end;

procedure EilWp (A,K:coef; X0,X1,V0:real; var Y:coef);
var S,n:integer; Sum:=real;
begin
  n:=round(A[-1]); Sum:=0;
  for S:=0 to n-1 do Sum:=sum+A[s]*Y[S+1];
  Y[n+1]:=-Sum/A[n];
  for S:=1 to n do
    Y[S] :=Y[S]+V0*(K[S]*X0+Y[S+1]);
  Y[0]:=K[0]*X1+Y[1]
end;

procedure UrGod;
var R1,I1,R2,I2,R3,I3,Zn,Wtau:real;
begin
  HorComp(B,0,W,R1,I1);
  HorComp(A,0,W,R2,I2);
  Zn:=Sqr(R2)+Sqr(I2);
  R3:=(R1*R2+I1*I2)/Zn;
  I3:=(R2*I1-R1*I2)/Zn;
  Wtau:=W*Wtau;
  R1:=Cos(Wtau); I1:=Sin(Wtau);
  X:=R3*R1+I3*I1; Y:=R1*I3-I1*R3;
end;

procedure SZ;
begin
  UrGod;
  if Dx<>0 then s:=x0+round(x/Dx)
    else s:=x0;
  if Dy<>0 then z:=y0-round(y/Dy)

```

```

                else z:=y0
end;

procedure PointGod;
const Kw=1.2;
var Mdx,Mdy,Md,Sp,Zp:integer;
    Wp:real;
begin
    Sp:=s;Zp:=z;Wp:=w;
    repeat
        W:=Wp+Dw; Sz;
        Mdx:=abs(S-Sp);Mdy:=abs(Z-Zp);
        if Mdx>Mdy then Md:=Mdx
            else Md:=Mdy;
        if Md<>1 then
            if Md<Min then Dw:=Dw*Kw
                else Dw:=Dw/Md;
        until (Md=1) or (W>Wk);
        if Md=1 then PutPixel(S,Z,c)
    end;

procedure Scan;
var s:integer;
begin
    Dw:=(Wk-Wn)/Nsc;
    Xmin:=0; Xmax:=0; Ymin:=0; Ymax:=0;
    for s:=0 to Nsc do
        begin
            W:=Wn+S*Dw; UrGod;
            if X<Xmin then Xmin:=X;
            if X>Xmax then Xmax:=X;
            if Y<Ymin then Ymin:=Y;
            if Y>Ymax then Ymax:=Y
        end;
    if Syst and (Xmin>-1) then Xmin:=-1;
    X0Y0(true);ClearDevice;SystCoor
end;

procedure Godo;
var J3:char;
begin
    PutA;Ou('0-Exit'); J3:='1';
    Dw:=(Wk-Wn)/Nsc; W:=Wn; Sz;
    PutPixel(S,Z,C); SetColor(C);
    repeat

```

```

    PointGod; Str(W,T20);
    Clear(450,0,GetMaxX-1,10);
    Outtextxy(450,0,'W='+T20);
    if KeyPressed then J3:=ReadKey
until (W>Wk) or (J3='0');
SetColor(15)
end;

```

```

Procedure Service;

```

```

    Var J3:char;
    begin
        repeat
            putA; Ou('0-exit, 1-Mark, 2-Net, 3-Num, 4-Inscr');
            J3:=readkey;
            case J3 of
                '1':Mark;
                '2':Net;
                '3':Num;
                '4':Inscr
            end
        until J3='0'
    end;

```

```

Procedure FormMas;{ Ht,Gt:coefL for object}

```

```

    var S,Z,Zt:integer;
        Y,Yp:real;
    begin
        FormArBr; SystZam(Br,Ar,Ksz); Ht[-1]:=L;
        Dt:=D/L; V0:=Dt/Ks; Ht[L+1]:=Dt; Gt:=Ht;
        for S:=0 to round(Ar[-1]) do My[s]:=0;
        Zt:=round(Taur/V0); Y:=0;
        for Z:=0 to Zt do
            begin
                Ht[z]:=0; Gt[z]:=0
            end;
        for Z:=Zt+1 to L do
            begin
                if (Br[-1]=0) and (Br[0]=1) and
                    (Ar[-1]=0) and (Ar[0]=1) and (Kob=1)
                then if Tc='H' then Ht[z]:=1
                    else
                        if (Z=Zt+1) or (Z=Zt+2) then Gt[z]:=1
                            else Gt[z]:=0
                else
                    begin

```

```

        Yp:=Y;
        for S:=1 to Ks do EilWp(Ar,Ksz,1,1,V0,My);
        Y:=My[0]; Ht[z]:=Y; Gt[z]:=(Y-Yp)/Dt
    end
end
end;

```

```

Procedure FormSC;
var S:integer; Y:real;
begin
    FormMas;
    Xmin:=0; Xmax:=D; Ymin:=0; Ymax:=0;
    for S:=0 to L do
        begin
            case Tc of
                'H': Y:=Ht[s];
                'G': Y:=Gt[s]
            end;
            if Y<Ymin then Ymin:=Y;
            if Y>Ymax then Ymax:=Y
        end;
        XOY0(false); ClearDevice; Systcoor;
        case Tc of
            'H': Graphic(Ht,C);
            'G': Graphic(Gt,C)
        end
    end;
end;

```

```

procedure InputOb;
var m:integer;
begin
    repeat
        putA;
        Ou('0-exit, 1-B(p), 2-A(p), 3-ChainZv, 4-Tau');
        Str(Tau:1:2,T10); Ts:='Tau='+T10;
        Info; J1:=readkey;
        case J1 of
            '1':InpPol('B',B);
            '2':InpPol('A',A);
            '3':repeat
                PutA;Ou('0-exit, 1-Kob, 2-q, 3-R, 4-TN');
                Str(Kob:1:3,T10); Ts:='Kob='+T10;
                Str(q,T10); Ts:=Ts+', q='+T10;
                Str(R,T10); Ts:=Ts+', R='+T10;
            until J1='4'
        end
    until J1='0'
end;

```



```

Info; J2:=readkey;
case J2 of
  '1':Our('Kob',Kob);
  '2':Oui('q',q);
  '3':Oui('R',R);
  '4':begin
      A[-1]:=0; A[0]:=1;
      B[-1]:=1; B[0]:=0; B[1]:=1;
      for S:=1 to q do UmnPol(A,B,A);
      B[0]:=1;
      for S:=1 to R do
          begin
              str(S,T10); putA;
              OutTextXY(150,0,'Enter T['+T10+'], N['+T10+']);
              GotoXY(1,1); Read(B[1],m);
              for Z:=1 to m do UmnPol(A,B,A);
              end;
              B[-1]:=0; B[0]:=Kob
          end
      end
      until J2='0';
  '4':Our('Tau',Tau)
end
until J1='0'
end;

```

```

procedure DemObj;
begin
  Syst:=false;
  repeat
    PutA;
    Ou('0-exit, 1-W(p), 2-h(t), 3-g(t), 4-W(j*w), 5-A(w)');
    J1:=readkey;
    case J1 of
      '1':begin
          OutPol(50,'B',B);
          OutPol(300,'A',A);
          Str(Tau:1:3,T10);
          OutTextXY(50,(4+round(B[-1]))*10,'Tau='+T10);
          end;
      '2','3':begin
          if J1='2' then Tc:='H'
            else Tc:='G';
          repeat
            putA;

```

```

Ou('0-exit, 1-D, 2-Ks, 3-c, 4-Sc, 5-Graphic, 6-Gromk');
  Str(D:1:1,T10);   Ts:='D'+T10;
  Str(Ks,T10);      Ts:=Ts+', Ks='+T10;
  Str(C,T10);       Ts:=Ts+', C='+T10;
  Info; J2:=ReadKey;
  case J2 of
    '1':Our('D',D);
    '2':Oui('Ks',Ks);
    '3':Oui('C',C);
    '4':FormSC;
    '5':begin
      FormMas;
      if Tc='H' then Graphic(Ht,C)
        else Graphic(Gt,C)
      end;
    '6':Service
  end
  until J2='0'
end;
'4':repeat
  putA;
Ou('0-ex, 1-Wn, 2-Wk, 3-Nsc, 4-C, 5-SC, 6-godo, 7-IndW');
  Str(Wn:1:3,T10);   Ts:='Wn'+T10;
  Str(Wk:1:1,T10);   Ts:=Ts+', Wk='+T10;
  Str(Nsc,T10);      Ts:=Ts+', Nsc='+T10;
  Str(C,T10);        Ts:=Ts+', C='+T10;
  Info; J2:=ReadKey;
  case J2 of
    '1':Our('Wn',Wn);
    '2':Our('Wk',Wk);
    '3':Oui('Nsc',Nsc);
    '4':Oui('C',C);
    '5':Scan;
    '6':Godo;
    '7':repeat
      PutA;
Ou('0-exit, 1-W, 2-C, 3-Point, 4-Inscr');
  Str(W:1:3,T10);   Ts:='W'+T10;
  Str(C,T10);      Ts:=Ts+', C='+T10;
  Info; J3:=Readkey;
  case J3 of
    '1':Our('W',W);
    '2':Oui('C',C);
    '3':begin
      Sz; Setcolor(c);

```

```

                                Circle(S,Z,3);
                                setcolor(15)
                                end;
                                '4':Inscr
                                end
                                until J3='0'
                                end {case J2}
                                until J2='0';
'5': repeat
    putA;
    ou('0-exit, 1-Wn, 2-Wk, 3-C, 4-A(w)');
    Str(Wn:1:3,T10); Ts:='Wn='+T10;
    Str(Wk:1:3,T10); Ts:=Ts+', Wk='+T10;
    Str(C,T10);      Ts:=Ts+', C='+T10;
    Info; J3:=readkey;
    Case J3 of
        '1':Our('Wn',Wn);
        '2':Our('Wk',Wk);
        '3':Oui('C',C);
        '4': begin
            Dw:=Wk/L; Mz[-1]:=L;
            ymin:=0; ymax:=0;
            for S:=0 to L do
                begin
                    W:=S*Dw;
                    if W<Wn
                        then
                            Mz[S]:=0
                        else
                            begin
                                UrGod;
                                y:=sqrt(sqr(x)+sqr(y));
                                Mz[S]:=y;
                                if y>ymax
                                    then ymax:=y
                                end
                            end
                        end;
                    X0Y0(false);
                    clear(0,15,getMaxX-1,getMaxY-15);
                    systcoor;graphic(Mz,C); J1:='4';
                    end;
                end {case J2}
                until J3='0';
            end
        end
    until J1='0'

```

```
end;

begin
  C:=15;    B[-1]:=0; B[0]:=1; A:=B; Kob:=1; Tau:=4;
  repeat
    PutA;
    Ou('Esc-ex, 1-Input, 2-DemOb');
    J:=readkey;
    Case J of
      '1':InputOb;
      '2':DemObj;
    end
  until J=#27;
  CloseGraph
end.
```

```

Program DemObsu;
  uses    Crt,Graph,Serv,Polynom,Groms;
  type    M2C=array[1..6,1..10] of coef;
          M2=array[1..6,-1..16] of real;
          Matr=array[1..12,1..13] of real;

  const   Ninp:integer=1;
          Nout:integer=1;
          D:real=20;
          Ks:integer=1;
          Min:integer=5;
          Max:integer=15;
          kw:real=1.5;
          n:integer=4;
          m:integer=3;
          Wn:real=0.01;
          Wk:real=2;
          Nsc:integer=50;
          Tgod:char='0';
          Eps:real=1e-6;

  var     Aa,Ba,Ksz,Mx,My,E,F:coef;
          A,B:M2;
          Rm:M2C; Ht,Hta,Gt,Gta:coef1;
          Dt,V0,W,DW:real;T20:string[20];
          Kornia, Kornib:Kor;
          NPB:integer;

procedure OutPol ( x:integer; Id:string; A:coef);
  var S,N:integer;
  begin
    N:=round(A[-1]);
    for S:=0 to N do
      begin
        Str(S,T10); Str(A[S],Ts);
        Outtextxy(X,(S+2)*10,id+'['+T10+']='+Ts)
      end
    end;

procedure SystUr (n:integer; A:Matr; var X:coef);
  const Eps=1e-9;
  var Zmax,z,s,i,j:integer;
      r:real;
      b:boolean;
  begin

```

```

b:=true;
if n=1
  then
    if abs(A[1,1])<Eps
      then b:=false
      else X[1]:=A[1,2]/A[1,1]
    else
      begin
        i:=1;
        while (i<=n-1) and b do
          begin
            Zmax:=1;
            for z:=i+1 to n do
              if abs(A[z,i])>abs(A[Zmax,i])
                then Zmax:=z;
            if abs(A[Zmax,i])<Eps
              then b:=false
              else
                begin
                  if Zmax>i then
                    for j:=i to n+1 do
                      begin
                        r:=a[i,j];
                        A[i,j]:=A[Zmax,j];
                        A[zmax,j]:=r
                      end;
                    for z:=i+1 to n do
                      if abs(A[z,i])>Eps then
                        begin
                          r:=A[z,i]/A[i,i];
                          for s:=i+1 to n+1 do
                            A[z,s]:=A[z,s]-r*A[i,s]
                          end
                        end;
                  end;
                inc(i)
              end;
            if b then
              if abs(A[n,n])<Eps then b:=false;
            if b then
              begin
                X[n]:=A[n,n+1]/A[n,n];
                for i:=n-1 downto 1 do
                  begin
                    r:=0;
                    for s:=i+1 to n do

```

```

                r:=r+A[i,s]*X[s];
                X[i]:=(A[i,n+1]-r)/A[i,i];
            end
        end
    end;
    if b then X[-1]:=n else X[-1]:=0
end;

procedure SystZam (B,A:coef; var K:coef);
var S,n,R,m:integer;
    Sum:real;
begin
    N:=round(A[-1]); m:=round(B[-1]);
    if m<n then
        for S:=m+1 to n do B[S]:=0;
    K[0]:=B[n]/A[n];
    for s:=1 to n do
        begin
            Sum:=0;
            for R:=0 to s-1 do
                Sum:=Sum+K[R]*A[n-s+R];
            K[S]:=(B[n-s]-Sum)/A[n]
        end
    end;
end;

procedure EilWp (A,K:coef; X0,X1,V0:real; var Y:coef);
var s,n:integer;
    Sum:real;
begin
    n:=round(A[-1]); Sum:=0;
    for s:=0 to n-1 do
        Sum:=Sum+A[s]*Y[S+1];
    Y[n+1]:=-Sum/A[n];
    for s:=1 to n do
        Y[s]:=Y[s]+V0*(K[s]*X0+Y[s+1]);
    Y[0]:=K[0]*X1+Y[1]
end;

procedure EilSu(A,B:M2; Mx0:coef; V0:real; var Y:coef);
var z,s,n,m:integer;
    Sum:real;
begin
    n:=round(A[1,-1]); m:=round(B[1,-1]);
    for z:=1 to n do
        begin

```

```

    Sum:=0;
    for s:=1 to n do Sum:=Sum+A[z,s]*Y[s];
    for s:=1 to m do Sum:=Sum+B[z,s]*Mx0[s];
    Y[z]:=Y[z]+V0*Sum;
end
end;

procedure RedWp (Eps:real; var B,A:coef);
var m,n,s,z:integer;
    U:boolean;
    E,F:coef;
    R,I,Epsf:real;
begin
    Epsf:=Eps*1e3;
    repeat
        KorPolMs (B,Eps,KorniB); m:=round(B[-1]);
        KorPolMs (A,Eps,KorniA); n:=round(A[-1]);
        s:=0; U:=false;
        repeat
            inc(s);
            if s<=m then
                begin
                    R:=KorniB[1,s]; I:=KorniB[2,s];
                    z:=0;
                    repeat
                        inc(z);
                        if (abs(R-KorniA[1,z])<=Epsf) and
                            (abs(I-KorniA[2,z])<=Epsf) then U:=true
                    until U or (z>=n);
                    if U then
                        begin
                            if abs(I)<=Epsf
                                then
                                    begin
                                        E[-1]:=1; E[0]:=-R; E[1]:=1
                                    end
                                else
                                    begin
                                        E[-1]:=2; E[0]:=R*R+I*I;
                                        E[1]:=-2*R; E[2]:=1
                                    end;
                            DelPol (B,E,B,F); DelPol (A,E,A,F)
                        end
                    end
                until U or (s>=m)

```



```

    until U=false
end;

procedure WpSu(A,B:M2);
var i,z,s:integer;
    E,F,C,D:coef;
begin
    n:=round(A[1,-1]); m:=round(B[1,-1]);
    for z:=1 to n do
        begin
            for s:=1 to n do
                if s=z
                then
                    begin
                        Rm[z,s,-1]:=1; Rm[z,s,0]:=A[z,z];
                        Rm[z,s,1]:=-1
                    end
                else
                    begin
                        Rm[z,s,-1]:=0; Rm[z,s,0]:=A[z,s]
                    end;
                for s:=1 to m do
                    begin
                        Rm[z,n+s,-1]:=0; Rm[z,n+s,0]:=-B[z,s]
                    end
                end;
            end;
        for i:=1 to n-1 do
            begin
                C:=Rm[i,i];
                for z:=i+1 to n do
                    if not((Rm[z,i,-1]=0)
                        and (Rm[z,i,0]=0))
                    then
                        begin
                            D:=Rm[z,i];
                            for s:=i+1 to n+m do
                                begin
                                    UmnPol(Rm[z,s],C,E);
                                    UmnPol(Rm[i,s],D,F);
                                    AlSumPol(E,F,'-',Rm[z,s]);
                                    RedPol(1e-9,Rm[z,s])
                                end
                            end
                        end
                    end;
            end;
        if Rm[n,n,n]<0 then

```

```

for s:=n to n+m do
  for i:=0 to n do Rm[n,s,i]:=-Rm[n,s,i];
for z:=n-1 downto 1 do
  for s:=1 to m do
    begin
      C[-1]:=0; C[0]:=0;
      for i:=z+1 to n do
        begin
          UmnPol(Rm[z,i],Rm[i,n+s],D);
          AlSumPol(C,D,'+',C)
        end;
      UmnPol(Rm[z,n+s],Rm[n,n],D);
      AlSumPol(D,C,'-',C);
      DelPol(C,Rm[z,z],Rm[z,n+s],D)
    end
  end;
end;

procedure UrGod;
  var X1:real;
      z,s:integer;
      Ms:Matr;
      R:Coef;
begin
  for z:=1 to n do
    begin
      for s:=1 to n do
        begin
          Ms[z,s]:=A[z,s];
          Ms[z+n,s+n]:=A[z,s];
          if s=z
            then
              begin
                Ms[z,s+n]:=w;
                Ms[z+n,s]:=-w
              end
            else
              begin
                Ms[z,s+n]:=0;Ms[z+n,s]:=0
              end
            end;
          Ms[z,2*n+1]:=-B[z,Ninp];
          Ms[z+n,2*n+1]:=0
        end;
      SystUr(2*n,Ms,R); x:=R[Nout]; y:=R[Nout+n];
    end;
end;

```

```

procedure SZ;
begin
  UrGod;
  if Dx<>0 then s:=x0+round(x/Dx)
    else s:=x0;
  if Dy<>0 then z:=y0-round(y/Dy)
    else z:=y0
end;

procedure LineGod;
var Md,Mdx,Mdy,Sp,Zp:integer;
    Wp,R:real;
begin
  Sp:=s;Zp:=z;Wp:=w;R:=(Min+Max)/2;
  repeat
    W:=Wp+Dw; Sz;
    Mdx:=abs(s-Sp); Mdy:=abs(z-Zp);
    if Mdx>Mdy then Md:=Mdx
      else Md:=Mdy;
    if (Md>Max) or (Md<Min) then
      if Md<Min then Dw:=Dw*kw
        else Dw:=Dw*R/Md
    until ((Md<=Max) and (Md>=Min))
      or (W>Wk);
    if (Md<=Max) and (Md>=Min)
      then Line(Sp,Zp,S,Z)
  end;

procedure Scan;
var s:integer;
begin
  Dw:=(Wk-Wn)/Nsc;
  Xmin:=0; Xmax:=0;Ymin:=0;Ymax:=0;
  for s:=0 to Nsc do
    begin
      W:=Wn+S*Dw; UrGod;
      if x<Xmin then Xmin:=x;
      if x>Xmax then Xmax:=x;
      if y<Ymin then Ymin:=y;
      if y>Ymax then Ymax:=y
    end;
  if (Tgod='N') and (Xmin>-1) then Xmin:=-1;
  X0Y0(true); ClearDevice; SystCoor

```

```

end;

procedure Godo;
var J3:char;
begin
  PutA; Ou('0-Exit'); J3:='1';
  Dw:=(Wk-Wn)/Nsc; W:=Wn; Sz;
  PutPixel(S,Z,C); SetColor(C);
  repeat
    LineGod; Str(W,T20);
    Clear(450,0,GetMaxX-1,10);
    Outtextxy(450,0,'W='+T20);
    if KeyPressed then J3:=ReadKey
  until (W>Wk) or (J3='0');
  SetColor(15)
end;

procedure FormSC(Tc:char);
var s,z:integer; y,yp,yr:real;
begin
  Dt:=D/L; V0:=Dt/Ks;
  Xmin:=0; Xmax:=D;
  Ymin:=0; Ymax:=0; y:=0; Mx[-1]:=m;
  Ht[-1]:=L; Ht[0]:=0; Ht[L+1]:=Dt;
  Gt[-1]:=L; Gt[0]:=0; Gt[L+1]:=Dt;
  for s:=1 to m do Mx[s]:=0;
  Mx[Ninp]:=1;
  for s:=1 to n do My[s]:=0;
  for z:=1 to L do
  begin
    yp:=y;
    for s:=1 to Ks do
      EilSu(A,B,Mx,V0,My);
    y:=My[Nout]; Ht[z]:=y;
    yr:=(y-yp)/dt; Gt[z]:=yr;
    if Tc='H' then yr:=y;
    if yr<Ymin then Ymin:=yr;
    if yr>Ymax then Ymax:=yr
  end;
  XOY0(false); ClearDevice; SystCoor;
end;

procedure FormHtGt(Tc:char);
var s,z:integer; y,yp,yr:real;
begin

```

```

dt:=D/L;
if Tc='H'
  then
    begin
      Ht[-1]:=L; Ht[0]:=0; Ht[L+1]:=dt;
      Hta[-1]:=L; Hta[0]:=0; Hta[L+1]:=dt
    end
  else
    begin
      Gt[-1]:=L; Gt[0]:=0; Gt[L+1]:=dt;
      Gta[-1]:=L; Gta[0]:=0; Gta[L+1]:=dt
    end;
repeat
  PutA;
  if Tc='H'
    then
      Ou('0-exit, 1-D, 2-Ks, 3-Ninp, 4-Nout, 5-C, 6-
hsu, 7-hwp')
    else
      Ou('0-exit, 1-D, 2-Ks, 3-Ninp, 4-Nout, 5-C, 6-
gsu, 7-gwp');
  Str(D:1:2,T10); Ts:='D='+T10;
  Str(Ks,T10); Ts:=Ts+', Ks='+T10;
  Str(Ninp,T10); Ts:=Ts+', Ninp='+T10;
  Str(Nout,T10); Ts:=Ts+', Nout='+T10;
  Str(C,T10); Ts:=TS+', C='+T10;
  Info; J1:=ReadKey;
  case J1 of
    '1':Our('D',D);
    '2':Oui('Ks',Ks);
    '3':Oui('Ninp',Ninp);
    '4':Oui('Nout',Nout);
    '5':Oui('C',C);
    '6':begin
      FormSC(Tc);
      if Tc='H' then Graphic(Ht,C)
        else Graphic(Gt,C)
      end;
    '7':begin
      FormSC(Tc);
      WpSu(A,B); Aa:=Rm[n,n]; RedPol(1e-9,Aa);
      Ba:=Rm[Nout,N+Ninp]; RedPol(1e-9,Ba);
      N:=round(Aa[-1]);
      V0:=Dt/Ks;
      SystZam(Ba,Aa,Ksz); y:=0;

```

```

    for s:=0 to n do My[s]:=0;
    for z:=1 to L do
        begin
            yp:=y;
            for S:=1 to ks do
                EilWp(Aa,Ksz,1,1,V0,My);
            y:=My[0];      Hta[z]:=y;
            yr:=(y-yp)/dt; Gta[z]:=yr;
            end;
            if Tc='H' then Graphic(Hta,C)
                else Graphic(Gta,C)
            end
        end
    end
until J1='0'
end;

procedure DemObj;
begin
    repeat
        PutA;
        Ou('0-exit, 1-W(p), 2-h(t), 3-g(t), 4-W(j*w)');
        J1:=ReadKey;
        case J1 of
            '1':begin
                WpSu(A,B);
                Ba:=Rm[Nout,N+Ninp]; RedPol(1e-6,Ba);
                Aa:=Rm[n,n]; RedPol(1e-6,Aa);
                RedWp(Eps,Ba,Aa);
                OutPol(50,'B',Ba); OutPol(300,'A',Aa);
            end;
            '2':repeat
                PutA; Ou('0-exit, 1-D, 2-Ks, 3-h(t)');
                Str(D:1:2,T10); Ts:='D='+T10;
                Str(Ks,T10); Ts:=Ts+', Ks='+T10;
                Info; J2:=ReadKey;
                case J2 of
                    '1':Our('D',D);
                    '2':Oui('Ks',Ks);
                    '3': begin
                            FormSC('H'); Graphic(Ht,15)
                        end
                end
            end
        until J2='0';
            '3':repeat
                PutA;

```

```

Ou('0-exit, 1-D, 2-Ks, 3-g(t)');
Str(D:1:2,T10); Ts:='D='+T10;
Str(Ks,T10); Ts:=Ts+', Ks='+T10;
Info; J2:=ReadKey;
case J2 of
  '1':Our('D',D);
  '2':Oui('Ks',Ks);
  '3':begin
      FormSC('G'); Graphic(Gt,15)
    end
  end
until J2='0';
'4':repeat
  PutA;
  Ou('0-ex, 1-Wn, 2-Wk, 3-Nsc, 4-AFCob');
  Str(Wn:1:3, T10); Ts:='Wn='+T10;
  Str(Wk:1:1, T10); Ts:=Ts+', Wk='+T10;
  Str(Nsc, T10); Ts:=Ts+', Nsc='+T10;
  Str(C,T10); Ts:=Ts+', C='+T10;
  Info; J2:=readkey;
  case J2 of
    '1':Our('Wn',Wn);
    '2':Our('Wk',Wk);
    '3':Oui('Nsc',Nsc);
    '4':begin
        Tgod:='0'; Scan; Cleardevice;
        Systcoor; Godo
      end;
  end
until J2='0'
end
until J1='0'
end;

```

```

begin
  n:=3; m:=3; C:=15;
  A[1,1]:=-1; A[1,2]:=0; A[1,3]:=0;
  A[2,1]:=2; A[2,2]:=-2; A[2,3]:=0;
  A[3,1]:=0; A[3,2]:=3; A[3,3]:=-3;
  B[1,1]:=1; B[1,2]:=0; B[1,3]:=0;
  B[2,1]:=0; B[2,2]:=2; B[2,3]:=0;
  B[3,1]:=0; B[3,2]:=0; B[3,3]:=3;
  A[1,-1]:=n; B[1,-1]:=m;
  repeat
    PutA;
  
```

```

Ou('Esc-exit, 1-Inp, 2-Ninp, 3-Nout, 4-DemOb, 5-
Serv');
Str(Ninp,T10); Ts:='Ninp='+T10;
Str(Nout,T10); Ts:=Ts+', Nout='+T10;
Info; J:=ReadKey;
case J of
  '1':repeat
    PutA;Ou('0-exit, 1-InpA, 2-InpB');
    J1:=readkey;
    case J1 of
      '1':begin
        Oui('n-degree of system',n);
        A[1,-1]:=n;
        for z:=1 to n do
          begin
            Str(z,T10);
            for s:=1 to n do
              begin
                Str(S,Ts);

Our('A['+T10+', '+Ts+'],' ,A[z,s])
                end
              end
            end;
          end;
        '2':begin
          Oui('m- number of inputs',m);
          B[1,-1]:=m;
          for z:=1 to n do
            begin
              Str(z,T10);
              for s:=1 to m do
                begin
                  Str(S,Ts);

Our('B['+T10+', '+Ts+'],' ,B[z,s])
                end
              end
            end
          end
        until J1='0';
      '2':repeat
        Oui('Ninp',Ninp)
        until Ninp<=m;
      '3':repeat
        Oui('Nout',Nout)

```



```
        until Nout<=n;
        '4':DemObj;
        '5':Service;
    end
    until J=#27;
    CloseGraph
end.
```

```

Program DemObGt;
  uses Graph,Crt,Serv,Groms;
  const
    D:real=30;
    Ds:real=40;
    Ks:integer=1;
    m: integer = 20;
    Wn:real=0.01;
    Wk:real=1.5;
    Nsc:integer=50;
    Max:integer=15;
    Min:integer=5;
    Kw:real=1.5;
    C:integer=10;
    Tg:char='L';
    k:real=2; T:real=5; Tau:real=1.5;
var
  Mg,Mh:Coef1; T20:string[20];
  w,dw,dt:real;
  Name:string[8]; f:file of real;

function INTMg(t:real):real;
  var S:integer;r:real;
  begin
    r:=t/Dt;S:=trunc(r);t:=frac(r);
    intMg:=Mg[s]+t*(Mg[s+1]-Mg[s])
  end;

procedure UrGod;
  var Wt,Dtw,g,x1:real;s,lw:integer;
  {x=Re(w),y=Im(w)}
  begin
    x:=0;y:=0;
    if W<Pi/(10*Dt)
    then
      begin
        for s:=1 to L-1 do
          begin
            Wt:=W*s*Dt;g:=Mg[s];
            x:=x+g*cos(wt);
            y:=y-g*sin(wt);
          end;
          x:=x*Dt;y:=y*Dt;
        end
      else
        begin

```

```

Dtw:=Pi/(10*w);
lw:=round(D/Dtw);
Dtw:=D/Lw;
for s:=1 to Lw-1 do
  begin
    Wt:=W*S*Dtw;
    g:=intMg(s*Dtw);
    x:=x+g*cos(wt); y:=y-g*sin(wt)
  end;
  x:=x*Dtw; y:=y*Dtw;
end;
end;

```

```

procedure Scan;
var s:integer;
begin
  Dw:=(Wk-Wn)/Nsc; Xu:=(getmaxx-Nsc) div 2; PutA;
  Xmin:=0; Xmax:=0; Ymin:=0; Ymax:=0;
  rectangle(Xu,0,Xu+Nsc,5); Ou('0-exit'); s:=0;
  repeat
    W:=Wn+s*Dw; UrGod; Line(Xu+s,1,Xu+s,4);
    if X<Xmin then Xmin:=X;
    if X>Xmax then Xmax:=X;
    if Y<Ymin then Ymin:=Y;
    if Y>Ymax then Ymax:=Y;
    if KeyPressed then J2:=ReadKey; inc(s)
  until (s>Nsc) or (J2='0');
  X0Y0(true);
end;

```

```

procedure SZ;
begin
  UrGod;
  if Dx<>0 then S:=X0+round(X/Dx)
    else S:=X0;
  if Dy<>0 then Z:=Y0-round(Y/Dy)
    else Z:=Y0
end;

```

```

procedure PointGod;
const Kw=1.2;
var Mdx,Mdy,Md,Sp,Zp:integer;
    Wp:real;
begin
  Sp:=s; Zp:=z; Wp:=w;

```

```

repeat
  W:=Wp+Dw; SZ;
  Mdx:=abs(S-Sp);Mdy:=abs(Z-Zp);
  if Mdx>Mdy then Md:=Mdx
    else Md:=Mdy;
  if Md<>1 then
    if Md<Min then Dw:=Dw*Kw
      else Dw:=Dw/Md;
  until (Md=1) or (W>Wk);
  if Md=1 then PutPixel(S,Z,c)
end;

```

```

procedure LineGod;
var Mdx,Mdy,Md,Sp,Zp:integer;
    Wp,r:real;
begin
  Sp:=s;Zp:=z;Wp:=w; r:=(Min+Max)/2;
  repeat
    W:=Wp+Dw; SZ;
    Mdx:=abs(S-Sp);Mdy:=abs(Z-Zp);
    if Mdx>Mdy then Md:=Mdx
      else Md:=Mdy;
    if (Md>Max) or (Md<Min) then
      if Md<Min then Dw:=Dw*Kw
        else Dw:=Dw/Md*r;
    until ((Md<=Max) and (Md>=Min)) or (W>Wk);
    if (Md<=Max) and (Md>=Min) then Line(Sp,Zp,S,Z)
  end;

```

```

procedure Godo;
var j3:char;
begin
  PutA;Ou('0-exit'); J3:='1'; dw:=(Wk-Wn)/Nsc;
  W:=Wn; SZ; PutPixel(S,Z,c); SetColor(c);
  repeat
    if Tg='P' then PointGod
      else LineGod;
    Str(W,T20);clear(450,0,getmaxx-1,10);
    outtextxy(450,0,'W='+T20);
    if keypressed then J3:=readkey
  until (W>Wk) or (J3='0'); SetColor(15)
end;

```

```

procedure AFCo;
begin

```

```

repeat
  PutA;
  Ou('0-exit, 1-Wn, 2-Wk, 3-Nsc, 4-Tg, 5-W(j*w) ' );
  Str(Wn:10:3,T10);Ts:='Wn='+T10;
  Str(Wk:10:3,T10);Ts:=Ts+', Wk='+T10;
  Str(Nsc,T10);Ts:=Ts+', Nsc='+T10;
  ts:=ts+', tg='+tg;
  Info; J1:=readkey;
  case j1 of
    '1':Our('Wn',Wn);
    '2':Our('Wk',Wk);
    '3':Oui('Nsc',Nsc);
    '4':if tg='P' then tg:='L'
          else tg:='P';
    '5':begin Scan;
          cleardevice;
          Systcoor;
          Godo end
  end
until J1='0'
end;

```

```

procedure ReadFl;
var e,s:integer;
    j3:char;
begin
  repeat
    repeat
      PutA;
      OutTextXY(150,0,'enter filename(length>1)');
      GotoXY(1,1); Readln(Name)
    until Length(Name)>1;
    Assign(f,Name); {$i-} Reset(f); {$i+} e:=IOResult;
    if e<>0
      then OU(' file is not found, press enter');
      readln
    until e=0;
    read(f,Mg[-1]); L:=round(Mg[-1]);
    for s:=0 to L do Read(f,Mg[s]);
    Read(f,dt); d:=dt*L; Close(f);
    puta; Ou('file has been read'); j3:=readKey
  end;

```

```

procedure WriteFl;
var s:integer;

```

```

begin
  PutA;
  OutTextXY(150,0,'enter filename(length>1)');
  GotoXY(1,1);
  repeat
    Read(Name)
  until Length(Name)>1;
  Assign(f,Name); Rewrite(f);
  x:=L; Write(f,X);
  for s:=0 to L do Write(f,Mg[s]);
  Write(f,dt); Close(f)
end;

```

```

procedure Imit;
var Ntau:integer; pst:real;j2:char;
begin
  repeat
    puta;
Ou('0-exit, 1-K, 2-T, 3-tau, 4-D, 5-FormMas, 6-WriteFl');
    Str(k:1:3,t10); ts:='K='+t10;
    Str(t:1:3,t10); ts:=ts+', T='+t10;
    Str(Tau:1:3,t10); ts:=ts+', Tau='+t10;
    Str(D:1:3,t10); ts:=ts+', D='+t10;
    info; J2:=readKey;
    case J2 of
      '1':Our('K',K);
      '2':Our('T',T);
      '3':Our('tau',tau);
      '4':Our('D',D);
      '5':begin
          dt:=D/L; Mg[-1]:=L; Ntau:=round(Tau/dt);
          puta; xu:=(Getmaxx-L) div 2;
          rectangle(xu,0,xu+L,5);
          for s:=0 to Ntau do
            begin Mg[s]:=0; Line(xu+s,1,xu+s,4) end;
          Ymin:=0; Ymax:=0;
          for s:=Ntau+1 to L do
            begin
              Line(Xu+s,1,xu+s,4);
              Pst:=(s-Ntau)*dt/T;
              if Pst<80 then Y:=K/T*exp(-Pst)
                else Y:=0;
              Mg[s]:=y;
              if Y<Ymin then Ymin:=y;
              if Y>Ymax then Ymax:=Y
            end;
          end;
    end;
  repeat

```

```

        end;
        Xmin:=0; Xmax:=D; X0Y0(false); ClearDevice;
        SystCoor; Graphic(Mg,C); Mg[L+1]:=dt
    end;
    '6':WriteFl
end
until J2='0'
end;

begin
repeat
    PutA;
    Ou('Esc-exit, 1-Input, 2-DemObj, 3-Service');
    J:=readKey;
    case J of
        '1':repeat
            puta;
            Ou('0-exit, 1-ReadFl, 2-InpMas, 3-Imit');
            J1:=ReadKey;
            case J1 of
                '1':ReadFl;
                '2':begin
                    Oui('L',L); Our('dt',dt);
                    Mg[-1]:=L; Mg[L+1]:=dt;
                    for s:=0 to L do
                        begin
                            Str(s,t10);
                            Our('Mg['+t10+']',Mg[s])
                        end
                    end;
                end;
                '3':Imit
            end
        until J1='0';
        '2':repeat
            puta;
            Ou('0-exit, 1-C, 2-h(t), 3-g(t), 4-Afco');
            Str(c,t10); ts:='C='+t10;
            info; J1:=readKey;
            case J1 of
                '1':Oui('C',c);
                '2':begin
                    Mh[-1]:=L; Mh[0]:=0; Mh[L+1]:=dt;
                    Ymin:=0; Ymax:=0;
                    for s:=1 to L do
                        begin

```

```

        y:=Mh[s-1]+(Mg[s]+Mg[s-1])/2*dt;
        Mh[s]:=y;
        if Y<Ymin then Ymin:=y;
        if y>Ymax then Ymax:=y
        end;
        Xmin:=0; Xmax:=D; X0Y0(false);
        ClearDevice; SystCoor; Graphic(Mh,c)
    end;
'3':begin
    Ymin:=0; Ymax:=0;
    for s:=0 to L do
        begin
            y:=Mg[s];
            if y<Ymin then Ymin:=y;
            if y>Ymax then Ymax:=y
            end;
            Xmin:=0; Xmax:=D;
            X0Y0(false); ClearDevice;
            SystCoor; Graphic(Mg,c)
        end;
    '4':Afco
    end
    until J1='0';
    '3':Service;
    end
    until J=#27;
    CloseGraph
End.

```



```

program DemObRe;
  uses Graph, Crt, Serv, Groms;
  const
    Syst:boolean=false;
    k:real=2;
    T:real=5;
    Tau:real=1.5;
    D:real=20;
    Wn:real=0.01;
    Wk:real=3;
    Nsc:integer=50;
    Nvr:integer=1;
    Min:integer=5;
    Max:integer=15;
    Kw:real=1.5;
    Tg:char='P'; {'L'}
  var
    W,Dw,Dt:real;
    Re,Im,Ht,Gt:CoefL;
    T20:string[20];

  procedure UrGod;
    var R1,J1,R2,J2,wT,wTau,Z:real;
    begin
      wT:=w*T; wTau:=w*Tau;
      z:=1+Sqr(wT);
      R1:=k/Z;      J1:=k*wT/z;
      R2:=cos(wTau); J2:=sin(wTau);
      x:=(R1*R2-J1*J2);
      y:=-(R1*J2+J1*R2)
    end;

  procedure SZ;
  begin
    UrGod;
    if Dx<>0 then S:=X0+round(x/Dx)
      else S:=X0;
    if Dy<>0 then Z:=Y0-round(Y/Dy)
      else Z:=Y0
  end;

  procedure Scan;
  var S:integer;
  begin
    Dw:=(Wk-Wn)/Nsc;

```

```

Xmin:=0; Xmax:=0; Ymin:=0; Ymax:=0;
for S:=0 to Nsc do
begin
  W:=Wn+S*Dw; UrGod;
  if X<Xmin then Xmin:=X;
  if X>Xmax then Xmax:=X;
  if Y<Ymin then Ymin:=Y;
  if Y>Ymax then Ymax:=Y;
end;
X0Y0(true);
ClearDevice;
SystCoor;
end;

procedure PointGod;
const Kw=1.2;
var Mdx,Mdy,Md,Sp,Zp:integer;
    Wp:real;
begin
  Sp:=S; Zp:=Z; Wp:=W;
  repeat
    W:=Wp+Dw; Sz;
    Mdx:=abs(S-Sp);
    Mdy:=abs(Z-Zp);
    if Mdx>Mdy then Md:=Mdx
      else Md:=Mdy;
    if Md<>1 then
      if Md<1 then Dw:=Dw*Kw
        else Dw:=Dw/Md;
  until (Md=1) or (w>Wk);
  if C>=0
    then PutPixel(S,Z,C)
    else PutPixel(S,Z,GetMaxColor-GetPixel(S,Z))
end;

procedure LineGod;
var Mdx,Mdy,Md,Sp,Zp:integer;
    Wp,R:real;
begin
  Sp:=S; Zp:=Z; Wp:=W; R:=(Min+Max)/2;
  repeat
    W:=Wp+Dw; Sz;
    Mdx:=abs(S-Sp); Mdy:=abs(Z-Zp);
    if Mdx>Mdy then Md:=Mdx
      else Md:=Mdy;

```

```

    if (Md>Max) or (Md<Min)
    then
        if Md<Min then Dw:=Dw*Kw
        else Dw:=Dw*R/Md
    until ((Md<=Max) and (Md>=Min)) or (W>Wk);
    if (Md<=Max) and (Md>=Min) then Line(Sp, Zp, S, Z);
    if C<0 then
        PutPixel(S, Z, GetMaxColor-GetPixel(S, Z))
end;
```

```
procedure Godo;
```

```

var J3:char;
begin
    PutA; Ou('0-exit'); J3:='1';
    Dw:=(Wk-Wn)/Nsc; W:=Wn; Sz;
    if C>=0 then SetColor(C)
        else SetWriteMode(1);
    PutPixel(S, Z, C);
    repeat
        case Tg of
            'P':PointGod;
            'L':LineGod;
        end;
        Str(w, T20); Clear(450, 0, GetMaxX-1, 10);
        OutTextXY(450, 0, 'w='+T20);
        if KeyPressed then J3:=ReadKey;
    until (w>wk) or (J3='0');
    if C>=0 then SetColor(15)
        else SetWriteMode(0)
end;
```

```
procedure AFReIm(Re, Im:CoefL; C:integer );
```

```

var s, L, U, v:integer;
    x, y:real;
begin
    L:=round(Re[-1]); Xmin:=0; Xmax:=0;
    Ymin:=0; Ymax:=0;
    for s:=0 to L do
        begin
            x:=Re[s]; y:=Im[s];
            if x<Xmin then Xmin:=x;
            if x>Xmax then Xmax:=x;
            if y<Ymin then Ymin:=y;
            if y>Ymax then Ymax:=y;
        end;
end;
```

```

X0Y0(true); ClearDevice; SystCoor;
if C>=0 then SetColor(C)
           else SetWriteMode(1);
Moveto(X0+round(Re[0]/Dx),Y0-round(Im[0]/Dy));
for S:=1 to L do
begin
  U:=X0+round(Re[S]/Dx);
  v:=Y0-round(Im[S]/Dy);
  LineTo(U,v);
  if C<0 then
    PutPixel(U,v, GetMaxColor-GetPixel(U,v));
end;
if C>=0 then SetColor(15)
           else SetWriteMode(0);
end;

procedure GtRe(Re:CoefL; D:real; var Gt:CoefL);
var z,s,L:integer;
    Dt,Dw,Sum:real;
begin
  L:=round(Re[-1]);
  Dw:=Re[L+1]; Dt:=D/L;
  Gt[-1]:=L; Gt[L+1]:=Dt;
  for z:=0 to L do
begin
  Sum:=(Re[0]+Re[L]*cos(L*Dw*z*Dt))/2;
  for s:=1 to L-1 do
    Sum:=Sum+Re[s]*cos(s*Dw*Z*Dt);
  Gt[z]:=2/pi*Sum*Dw
end
end;

procedure GtIm(Im:CoefL; D:real; var Gt:CoefL);
var z,s,L:integer;
    Dt,Dw,Sum:real;
begin
  L:=round(Im[-1]); Dw:=Im[L+1];
  Dt:=D/L; Gt[-1]:=L; Gt[L+1]:=Dt;
  for z:=0 to L do
begin
  Sum:=Im[L]*Sin(L*Dw*z*Dt)/2;
  for s:=1 to L-1 do
    Sum:=Sum+Im[s]*sin(s*Dw*z*Dt);
  Gt[z]:=-2/pi*Sum*Dw
end
end;

```

```

end;

procedure HtGt(Gt:coefL; var Ht:coefL);
var L,s:integer; Dt:real;
begin
  L:=round(Gt[-1]); Dt:=Gt[L+1];
  Ht[-1]:=L; Ht[L+1]:=Dt; Ht[0]:=0;
  for s:=1 to L do
    Ht[s]:=Ht[s-1]+(Gt[s-1]+Gt[s])/2*Dt
  end;
end;

```

```

procedure ImitGt;
var y,Dt,St:real; Zt,s:integer;
begin
  Gt[-1]:=L; Dt:=D/L; Gt[L+1]:=Dt;
  Zt:=round(Tau/Dt); Xmin:=0; Xmax:=D;
  Ymin:=0; Ymax:=0;
  for s:=0 to L do
    begin
      if s<Zt
      then Gt[s]:=0
      else
        begin
          St:=(s-Zt)*Dt/T;
          if St<80 then y:=k/T*exp(-St)
          else y:=0;
          Gt[s]:=y;
          if y<Ymin then Ymin:=y;
          if y>Ymax then Ymax:=y
        end;
      end;
    X0Y0(false); ClearDevice; SystCoor;
    Graphic(Gt,15)
  end;
end;

```

```

procedure ImitHt;
var Zt:integer; St,Dt:real;
begin
  Ht[-1]:=L; Dt:=D/L; Ht[L+1]:=Dt;
  Zt:=round(Tau/Dt);
  for s:=0 to L do
    if s<Zt
    then Ht[s]:=0
    else
      begin

```

```

        St:=(s-Zt)*Dt/T;
        if St<80
            then Ht[s]:=k*(1-exp(-St))
            else Ht[s]:=k
        end;
    Xmin:=0; Xmax:=D; Ymin:=0; Ymax:=k;
    X0Y0(false); ClearDevice; SystCoor;
    Graphic(Ht,15)
end;

procedure ImitGodo;
var Cp:integer;
begin
    Nvr:=1; Scan; Cp:=C; C:=15; Godo; C:=Cp
end;

procedure FormReIm;
var Dw:real; s:integer;
begin
    Dw:=Wk/L; Re[-1]:=L; Re[L+1]:=Dw;
    Im[-1]:=L; Im[L+1]:=Dw; Nvr:=1;
    for s:=0 to L do
        begin
            w:=s*Dw; UrGod;
            Re[s]:=x; Im[s]:=y
        end
    end;
end;

procedure Imit;
begin
    repeat
        putA;
        Ou('0-exit,1-InpOB,2-D,3-g(t),4-h(t),5-Wn,6-Wk,
            7-NSC,8-Wjw,9-A(w)');
        str(D:1:3,T10); Ts:='D='+T10;
        str(Wn:1:3,T10); Ts:=Ts+', Wn='+T10;
        str(Wk:1:3,T10); Ts:=Ts+', Wk='+T10;
        str(Nsc,T10); Ts:=Ts+', Nsc='+T10;
        Info; J1:=ReadKey;
        case J1 of
            '1':repeat
                PutA; Ou('0-exit,1-K,2-T,3-Tau' );
                str(K:1:3,T10); Ts:=' K='+T10;
                str(T:1:3,T10); Ts:=Ts+', T='+T10;
                str(Tau:1:3,T10); Ts:=Ts+', Tau='+T10;
            end;
        end;
    until J1='0';
end;

```

```

        Info; J2:=ReadKey;
        case J2 of
            '1': Our ('K',K);
            '2': Our ('T',T);
            '3': Our ('Tau',Tau);
        end;
        until J2='0';
    '2':Our('D',D);
    '3':ImitGt;
    '4':ImitHt;
    '5':Our('Wn',Wn);
    '6':Our('Wk',Wk);
    '7':Oui('Nsc',Nsc);
    '8':ImitGodo;
    '9':begin
        Ht[-1]:=L;
        FormReIm;
        for s:=0 to L do
            Ht[s]:=sqrt(sqr(Re[s])+sqr(Im[s]));
            Xmin:=0; Xmax:=Wk;
            Ymin:=0; Ymax:=K;
            X0Y0(false); ClearDevice;
            SystCoor; Graphic(Ht,15);
        end
    end
until J1='0';
FormReIm
end;

begin
repeat
    PutA;
Ou('Esc-exit,1-Imit,2-c,3-gtRe,4-gtIm,5-W(jw),6-Serv');
Str(C,T10); Ts:='C'+T10;
Info; J:=ReadKey;
case J of
    '1':Imit;
    '2':Oui('C',C);
    '3':begin
        GtRe(Re,D,Gt);
        Graphic(Gt,C);
    end;
    '4':begin
        GtIm(Im,D,Gt); Graphic(Gt,C);
    end;
end;
end;
end;

```

```
    '5':AFcreIm(Re, Im, C);  
    '6':Service;  
end;  
until J=#27;  
CloseGraph;  
end.
```


Литература

1. Аверіна Т. В., Кубрак Н. А. Динаміка елементів систем: Навчальний посібник. – К.: ІЗМН, 1998.- 224с.
2. Бесекерский В.А., Попов Е.П. Теория систем автоматического регулирования. – М.: Наука, 1972.- 768с.
3. Дейч А.М. Методы идентификации динамических объектов. – М.: Энергия, 1979.- 240с.
4. Жученко А.І., Кваско М.З., Кубрак Н.А. Ідентифікація динамічних характеристик. Комп'ютерні методи. – К.: вир. відділ КЛТКМ та М, 2000. – 182с.
5. Кваско М.З., Кубрак Н.А. Динамічні моделі типових теплообмінних апаратів: Навч. посібник. – К.: ІЗМН, 1999. – 136с.
6. Кваско М.З., Кубрак А.І., Жученко А.І. Числові методи комп'ютерного моделювання автоматичних систем. Алгоритми і програми: Навч. посібник –К.: “Політехніка”,2003. – 360с.
7. Конторович М.И. Операционные исчисления и процессы в электрических цепях. – М.: Наука, 1964. – 328с.
8. Корнилов Ю.Г. Теоретические основы автоматического регулирования. – К.: 1966. – 397с.
9. Кубрак А.І. Ідентифікація динамічних характеристик елементів систем керування. Ч.1. Математичні методи: Навч. посібник. К.: ІСДО, 1995. – 208с.
10. Кубрак А.И., Жученко А.И., Кваско М.З. Комп'ютерне моделювання та ідентифікація автоматичних систем: Навч. посібник. –К.: ІВЦ “Видавництво «Політехніка»”, 2004. – 424с.
11. Кубрак А.И., Жученко А.И., Ярощук Л.Д. Численные методы и программы при исследовании автоматических смтем: Учеб. Пособие. – К.: УМК ВО, 1988. – 207с.

12. Кубрак А.І., Ярощук Л.Д., Аверіна Т.В. Програмування на Турбо Паскалі. Початковий курс: Навч. посібник. – К.: НМЦ ВО, 2000. – 188с.
13. Ладиев Р.Я., Остапенко Ю.А., Кубрак А.И., Кваско М.З. Аналитические методы описания объектов с сосредоточенными параметрами. –К.: КПИ, 1973. – 108с.
14. Мартыненко В.С. Операционное исчисление.-К.: Изд. Киевск. Университета, 1965.-185с.
15. Остапенко Ю.О. Ідентифікація і моделювання технологічних об'єктів керування. – К.: „Задруга”, 1999. – 420с.
16. Остапенко Ю.О. Типове АСР химко-технологических процессов: Учебн. пособие. – К.: УМК ВО, 1988. – 172с.
17. Ротач В.Я. Теория автоматического управления теплоэнергетическими процессами: Учебник для ВУЗов. – М.: Энергоатомиздат, 1985. – 296с.
18. Фаронов В.В. Турбо Паскаль 7.0. Начальный курс: Учебное пособие. – М.: Нолидж, 1997. – 616с.
19. Шостак Р.Я. Операционное исчисление. –М.: Высшая школа, 1968. – 191с.
20. Шульгин Ю.В. Автоматизация тепловых процес сов и установок. Ч.1. Теоретические основы автоматического регулирования тепловых процес сов: Учебн. пособие. –Харьков: НТУ “ХПИ”, 2002, – 96с.

Содержание

Введение	3
1. Преобразование Лапласа и его использование при исследовании автоматических систем	5
1.1. Определения преобразования Лапласа и границы его применения	6
1.2. Изображение элементарных функций	9
1.3. Основные свойства преобразования Лапласа	12
1.4. Передаточные функции, переходные, импульсные и частотные характеристики	17
1.5. Элементарные динамические звенья, их дифференциальные уравнения и передаточные функции	24
1.6. Переходные характеристики элементарных звеньев	28
1.7. Частотные характеристики элементарных звеньев	40
1.8. Определение корней полинома для обратного преобразования Лапласа дробно-рациональных изображений	53
1.9. Объединение массивов корней полиномов и редактирование результирующего массива	59
1.10. Алгебраические операции над полиномами с комплексными коэффициентами	63
1.11. Определение коэффициентов разложения дробно-рационального изображения на элементарные слагаемые	69
2. Аналитические модели простейших технологических объектов и элементов систем управления	73
2.1. Математическая модель многоемкостного теплового объекта типа проточной емкости	75
2.2. Моделирование объектов с регулированием давления пара или газа	87
2.3. Моделирование объектов с регулированием уровня жидкости	91
2.4. Объекты с регулированием концентрации	93
2.5. Динамика механических систем	95
2.6. RLC-цепочки	99

3. Компьютерные методы исследования динамики объектов с сосредоточенными параметрами	103
3.1. Динамика объектов, задаваемых системой обыкновенных дифференциальных уравнений	104
3.1.1. Определение передаточных функций	105
3.1.2. Расчет переходной характеристики	110
3.1.3. Расчет частотных характеристик	113
3.1.4. Расчет коэффициентов ряда Маклорена	116
3.2. Характеристики канала, задаваемого дробно-рациональной передаточной функцией	119
3.2.1. Расчет переходной характеристики	120
3.2.2. Расчет частотных характеристик	122
3.2.3. Вычисление коэффициентов ряда Маклорена	124
3.3. Характеристики канала, задаваемого массивом ординат импульсной характеристики	125
3.3.1. Расчет переходной характеристики	126
3.3.2. Расчет частотных характеристик	127
3.3.3. Вычисление коэффициентов ряда Маклорена	130
3.4. Динамика канала по его частотным характеристикам	132
3.4.1. Расчет переходной и импульсной характеристик	133
3.4.2. Вычисление коэффициентов ряда Маклорена	136
Приложения	140
Литература	177

Жученко Анатолий Иванович
Кубрак Наталия Анатольевна
Голинко Игорь Михайлович

ДИНАМИКА ОБЪЕКТОВ С СОСРЕДОТОЧЕННЫМИ ПАРАМЕТРАМИ