

МИНИСТЕРСТВО ВЫСШЕГО И СРЕДНЕГО СПЕЦИАЛЬНОГО
ОБРАЗОВАНИЯ РЕСПУБЛИКИ УЗБЕКИСТАН

Ташкентский Институт Текстильной и Легкой
Промышленности

Кафедра
«Автоматизация и компьютеризация технологических процессов»

МЕТОДИЧЕСКОЕ УКАЗАНИЕ
к лабораторным работам по курсу

«ЦИФРОВЫЕ СИСТЕМЫ АВТОМАТИЗАЦИИ И УПРАВЛЕНИЯ»

Ташкент - 2011

Данное методическое указание предназначено для бакалавров специальности 5521800 - «Автоматизация и управления». Рассмотрены темы занятий по курсу «Цифровые системы автоматизации и управления». Кратко приведены основные теоретические положения, касающиеся данных работ и переданы варианты заданий студентам для самостоятельной работы.

По данному методическому указанию можно изучить предмет «Цифровые системы автоматизации и управления» при подготовке бакалавров.

Цель настоящих лабораторных работ – получение студентами знаний и навыков по основам использования современных цифровых систем автоматизации и управления, а также развитие и закрепление навыков анализа дискретных систем.

Составители: к.т.н., доц. И.Х. Сиддиков,
ст. пр. Д.А. Холматов,
асс. Ю.А. Жукова.

Рецензенты: к.т.н., доц. Абдуллаев М. М. (ТГТУ)
к.т.н., доц. Ибрагимов Э.У. (ТИТЛП)

Рассмотрено и утверждено на заседании учебно-методического совета ТИТЛП

Протокол № _____ от _____ 20__ года

Лабораторная работа № 2

Цифровая реализация непрерывного регулятора

Цели работы: освоение методов переоборудования непрерывных регуляторов для реализации на цифровом компьютере

Задачи работы

- познакомиться с методами переоборудования непрерывных регуляторов в MATLAB
- научиться моделировать системы с цифровыми регуляторами
- научиться выбирать интервал квантования

Краткие теоретические сведения

Задача переоборудования

Классические методы синтеза позволяют построить непрерывный регулятор, который описывается передаточной функцией или соответствующим дифференциальным уравнением. Для непосредственной реализации такого регулятора можно использовать различные аналоговые элементы, в том числе и электронные (RC-цепочки, операционные усилители и т.п.).

В то же время в современных системах управления законы регулирования чаще всего реализуются с помощью бортового цифрового компьютера или микроконтроллера. При этом

- отсутствует дрейф параметров элементов регулятора (их изменение со временем, при изменении давления, температуры и т.п.);
- компьютер позволяет реализовывать достаточно сложные алгоритмы управления, в том числе и адаптивные, что крайне трудно сделать на базе аналоговой техники;
- для перестройки алгоритма управления не требуется менять аппаратуру, а надо просто заменить программу обработки данных.

Цифровые вычислительные устройства представляют собой элементы дискретного действия. Они периодически выполняют измерения входного сигнала и расчет нового управляющего воздействия. Период, через который эти операции повторяются, называется интервалом квантования и обозначается через T .

На цифровых компьютерах нельзя напрямую реализовать управляющее устройство, которое описывается дифференциальным уравнением в непрерывном времени. **Задача переоборудования** состоит в том, чтобы заменить спроектированный непрерывный регулятор цифровым устройством так, чтобы сохранить все существенные свойства непрерывной системы (устойчивость, качество, подавление постоянных возмущений).

Компьютер в контуре управления

Компьютер в контуре управления можно представить в виде последовательное соединение трех звеньев:

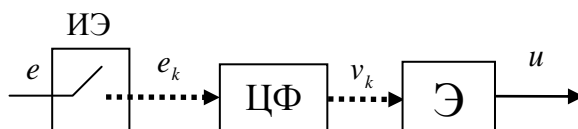
1) **импульсного элемента**, который выбирает из непрерывного сигнала ошибки $e(t)$ значения $e_k = e(kT)$ в моменты квантования $t_k = kT$ (при целых k); импульсный элемент моделирует аналого-цифровой преобразователь (АЦП);

2) линейного **цифрового фильтра**, который преобразует дискретную последовательность e_k в управляющую последовательность v_k ; свойства этого фильтра определяют закон управления;

3) восстанавливающее устройство (**экстраполятор**), которое восстанавливает непрерывный сигнал управления $u(t)$ из последовательности v_k ; экстраполятор моделирует цифро-аналоговый преобразователь (ЦАП), чаще всего используется **фиксатор нулевого порядка**, который удерживает постоянное значение $u(t)$ в течение очередного интервала квантования:

$$u(kT + \varepsilon) = v_k, \quad 0 \leq \varepsilon < T.$$

Схема цифрового регулятора показана на рисунке:

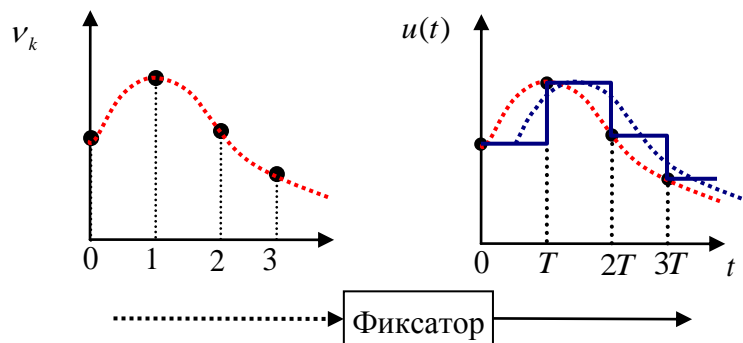


Блок **ИЭ** обозначает импульсный элемент (АЦП), блок **ЦФ** – цифровой фильтр, блок **Э** – экстраполятор (ЦАП). Точечные линии обозначают дискретные сигналы, сплошные линии – непрерывные сигналы.

Экстраполятор

Экстраполятором называют устройство, которое восстанавливает непрерывный сигнал управления $u(t)$ по дискретной последовательности значений $\{v_k\}$, поступающих с выхода цифрового фильтра в моменты времени $t_k = kT$. Обычно вычислительное запаздывание (время, необходимое на расчет очередного значения v_k) включают в модель объекта управления, поэтому считается, что цифровой фильтр выполняет обработку данных мгновенно.

В простейшем случае ЦАП, получив новый управляющий сигнал v_k от цифрового фильтра, просто удерживает (фиксирует) его в течение интервала квантования T (до получения следующего значения v_{k+1}). Такой экстраполятор называется **фиксатором нулевого порядка** (англ. *zero-order hold, ZOH*).



Фиксатор нулевого порядка

Фиксатор нулевого порядка восстанавливает сигнал по правилу

$$u(kT + \varepsilon) = v_k, \quad 0 \leq \varepsilon < T.$$

Здесь ε – так называемое локальное время, прошедшее с момента последнего срабатывания импульсного элемента.

Предположим, что дискретная последовательность получена в результате квантования некоторого непрерывного сигнала. Восстановленный сигнал будет представлять собой «ступеньки», высота которых совпадает с истинным значением сигнала в начале интервала. Если провести линию через середины этих «ступенек», получается сигнал, смещенный относительно исходного на $T/2$. Поэтому говорят, что последовательно примененные операции квантования и восстановления сигнала с помощью фиксатора нулевого порядка приводят к его **запаздыванию на половину периода**.

Существуют и более сложные экстраполяторы, учитывающие несколько последних значений последовательности $\{v_k\}$, но они используются на практике крайне редко из-за проблем в реализации.

Цифровые фильтры

Цифровой фильтр – это устройство, преобразующее входную дискретную последовательность

$$\{e_k\} = e_0, e_1, e_2, \dots$$

в выходную

$$\{v_k\} = v_0, v_1, v_2, \dots$$

В реальных фильтрах для расчета очередного значения управляющей последовательности v_k в момент времени $t = kT$ используется конечное число прошлых значений входного и выходного сигналов, хранящихся в оперативной памяти:

$$v_k = \mathfrak{Z}(e_k, e_{k-1}, \dots, e_{k-n}, v_{k-1}, \dots, v_{k-n})$$

Здесь $\mathfrak{Z}(\cdot)$ – некоторая функция своих переменных и n – целое число, называемое **порядком** фильтра. Чаще всего используют **линейные** законы управления, которые описываются формулой

$$v_k = a_0 e_k + a_1 e_{k-1} + \dots + a_n e_{k-n} - b_1 v_{k-1} - b_2 v_{k-2} - \dots - b_n v_{k-n},$$

где a_i ($i = 0, \dots, n$) и b_i ($i = 1, \dots, n$) – вещественные числа. Уравнение такого вида называют **линейным разностным уравнением** регулятора. Оно аналогично дифференциальному уравнению непрерывной системы, но входной и выходной сигналы изменяются в дискретном времени, т.е., определены только в моменты времени $t_k = kT$.

Как и для линейных непрерывных систем, для описания линейных цифровых фильтров (линейных дискретных систем) можно использовать операторный метод. В литературе чаще всего используется **оператор сдвига вперед** z

$$z e_k = e_{k+1}, \quad z^m e_k = e_{k+m}.$$

Обозначив через z^{-1} обратный оператор, получим $z^{-1} e_k = e_{k-1}$ и $z^{-m} e_k = e_{k-m}$. Тогда, перенося в левую часть все члены, зависящие от выходной последовательности, можно записать уравнение регулятора в операторной форме

$$(1 + b_1 z^{-1} + b_2 z^{-2} + \dots + b_n z^{-n}) v_k = (a_0 + a_1 z^{-1} + a_2 z^{-2} + \dots + a_n z^{-n}) e_k.$$

Отношение

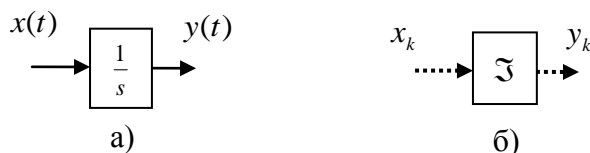
$$D(z) = \frac{a_0 + a_1 z^{-1} + a_2 z^{-2} + \dots + a_n z^{-n}}{1 + b_1 z^{-1} + b_2 z^{-2} + \dots + b_n z^{-n}} = \frac{a_0 z^n + a_1 z^{n-1} + a_2 z^{n-2} + \dots + a_n}{z^n + b_1 z^{n-1} + b_2 z^{n-2} + \dots + b_n}$$

называется **дискретной передаточной функцией** цифрового фильтра. Таким образом, в операторной форме получаем $v_k = D(z) e_k$.

Фактически задача переоборудования сводится к тому, чтобы заменить передаточную функцию $C(s)$ непрерывного регулятора дискретной передаточной функцией цифрового фильтра $D(z)$ так, чтобы сохранить все существенные свойства системы.

Методы численного интегрирования

Простейшие методы переоборудования основаны на приближенной замене интегрирующего звена с передаточной функцией $1/s$ его дискретной моделью. Это позволяет получить дискретную передаточную функцию цифрового регулятора, сделав соответствующую замену в передаточной функции непрерывного регулятора $C(s)$.



Непрерывный и дискретный интеграторы

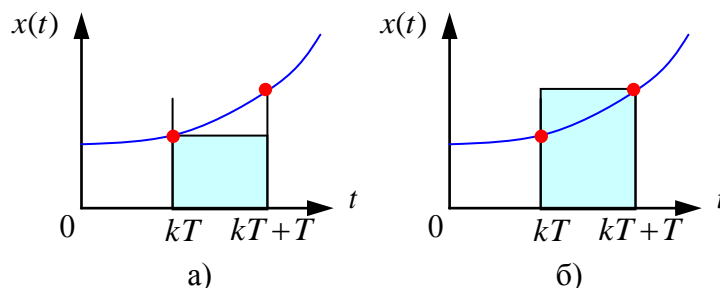
Пусть $x(t)$ и $y(t)$ — входной и выходной сигнал непрерывного интегратора. Если известно значение $y_k = y(kT)$, то

$$y_{k+1} = y(kT + T) = y_k + \int_{kT}^{kT+T} x(t) dt.$$

Такое звено приближенно заменяется дискретным интегратором, для которого

$$y_{k+1} = \mathfrak{Z}(y_k, y_{k-1}, \dots, x_{k+1}, x_k, \dots),$$

где $\mathfrak{Z}(\cdot)$ — некоторое правило построения следующего значения выхода по предыдущим значениям входа и выхода. Для решения этой задачи можно использовать любой метод численного интегрирования. Мы рассмотрим методы прямоугольников и трапеций.



Метод Эйлера (а) и метод обратных разностей (б)

При использовании метода Эйлера имеем

$$y_{k+1} = y_k + T x_k.$$

Используя оператор z (сдвиг вперед), получаем

$$z y_k = y_k + T x_k \Rightarrow \frac{y_k}{x_k} = \frac{T}{z-1}.$$

Таким образом, переоборудование по методу Эйлера сводится к замене

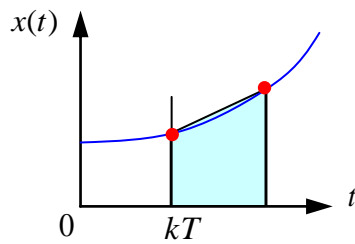
$$s \leftarrow \frac{z-1}{T}.$$

Аналогично можно построить правило замены для метода обратных разностей:

$$s \leftarrow \frac{z-1}{zT}.$$

Из курса численных методов известно, что методы прямоугольников дают низкую точность. Более совершенен метод трапеций:

$$y_{k+1} = y_k + \frac{T}{2}(x_{k+1} + x_k).$$



Метод трапеций

Формула интегрирования по методу трапеций приводит к замене

$$s \leftarrow \frac{2}{T} \cdot \frac{z-1}{z+1},$$

которая называется **преобразованием Тастина** (или Тустена).

Для повышения точности аппроксимации можно использовать более сложные методы, например, замены

$$s \leftarrow \frac{3}{T} \cdot \frac{z^2-1}{z^2+4z+1},$$

$$s \leftarrow \frac{10}{T} \cdot \frac{z^4+z^3-z-1}{z^4+6z^3+6z^2+6z+1},$$

соответствующие методам интегрирования Симпсона и Уэддля. Их главный недостаток состоит в том, что порядок переоборудованного регулятора будет выше, чем порядок непрерывного.

Переоборудование ПИД-регулятора

Рассмотрим непрерывный ПД-регулятор с передаточной функцией

$$C_{pd}(s) = K_c \left(1 + \frac{T_s s}{T_v s + 1} \right).$$

Дискретизация с помощью методов Эйлера, обратных разностей и Тастина дает дискретные регуляторы вида

$$D_{pd}(z) = \frac{K_c [a_1 z + a_0]}{b_1 z + b_0},$$

где коэффициенты равны

метод Эйлера

$$a_1 = T_s + T_v, \quad a_0 = T - (T_s + T_v), \quad b_1 = T_v, \quad b_0 = T - T_v.$$

метод обратных разностей

$$a_1 = T + (T_s + T_v), \quad a_0 = -(T_s + T_v), \quad b_1 = T + T_v, \quad b_0 = -T_v.$$

преобразование Тастина

$$a_1 = T + 2(T_s + T_v), \quad a_0 = T - 2(T_s + T_v), \quad b_1 = T + 2T_v, \quad b_0 = T - 2T_v.$$

Все регуляторы имеют тот же самый порядок (равный 1), что и непрерывный регулятор. Полученные дискретные регуляторы только приближенно заменяют непрерывный, фактически они всегда будут работать несколько хуже, чем $C(s)$.

ПД-регулятор будем переоборудовать с помощью преобразования Тастина (интегрирования методом трапеций), которое является наиболее точным из этих методов. В системе MATLAB для этого можно использовать функцию **c2d** из пакета Control Toolbox:

>> Dpd = c2d (Cpd, T, 'tustin')

Здесь **Cpd** – модель (например, передаточная функция) непрерывного ПД-регулятора, **T** – интервал квантования.

Теперь рассмотрим интегральный канал:

$$C_I(s) = \frac{1}{T_I s}.$$

Используя рассмотренные выше методы переоборудования, получаем

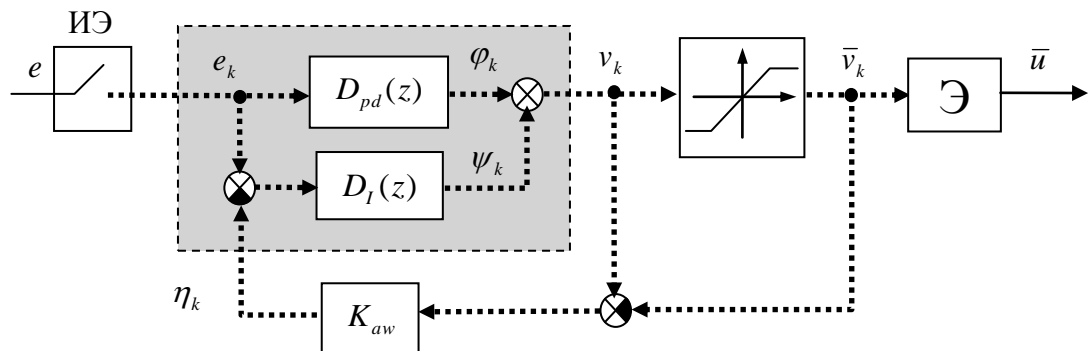
метод Эйлера
$$D_I(z) = \frac{T}{T_I(z-1)},$$

метод обратных разностей
$$D_I(z) = \frac{Tz}{T_I(z-1)},$$

преобразование Тастина
$$D_I(z) = \frac{T(z+1)}{2T_I(z-1)}.$$

Как будет показано дальше, для переоборудования интегрального канала лучше использовать преобразование Эйлера.

Ниже показана схема цифрового регулятора с компенсацией насыщения:



Здесь сплошные линии обозначают непрерывные сигналы, а штриховые – дискретные (числовые последовательности). ИЭ обозначает импульсный элемент (АЦП), а блок Э – экстраполятор (ЦАП).

Алгебраические циклы

Пусть интегральный канал переоборудован по методу обратных разностей

$$D_I(z) = \frac{Tz}{T_I(z-1)},$$

что соответствует разностному уравнению

$$\psi_k = \psi_{k-1} + \frac{T}{T_I}(e_k - \eta_k). \quad (*)$$

Теперь построим выражение для сигнала $\eta_k = K_{aw}(v_k - \bar{v}_k)$. Учтем, что

$$v_k = \varphi_k + \psi_k \quad \text{и} \quad \bar{v}_k = \text{sat}(v_k) = \text{sat}(\varphi_k + \psi_k),$$

где функция $\text{sat}(\cdot)$ задает нелинейность типа «насыщение»:

$$\text{sat}(v_k) = \begin{cases} u_{\min}, & v_k < u_{\min}, \\ v_k, & u_{\min} \leq v_k \leq u_{\max}, \\ u_{\max}, & v_k > u_{\max}. \end{cases}$$

Объединяя эти формулы, получим разностное уравнение для вычисления ψ_k :

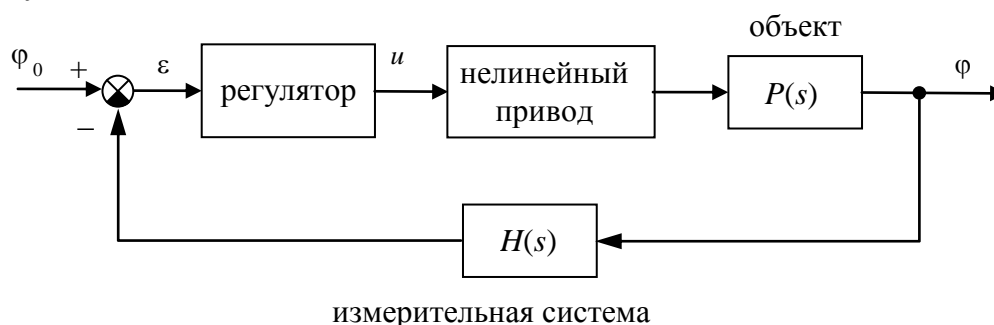
$$\psi_k = \psi_{k-1} + \frac{T}{T_I} [e_k - K_{aw}(\varphi_k + \psi_k - \text{sat}(\varphi_k + \psi_k))].$$

В этой формуле значение ψ_k , которое требуется рассчитать, входит и в правую часть! Это значит, что для вычисления ψ_k требуется не просто подставить в формулу известные значения, а решить нелинейное уравнение относительно ψ_k . Такое явление называется **алгебраическим циклом**, его желательно избегать. Более того, в сложных случаях это уравнение может не иметь решения вообще. Система MATLAB-SIMULINK выдает предупреждение в случае обнаружения алгебраического цикла (*algebraic loop*) при моделировании.

Для того, чтобы не было алгебраического цикла, правая часть разностного уравнения (аналогичного уравнению (*)) не должна зависеть от ψ_k . Это будет в том случае, если передаточная функция $D_I(z)$ – строго правильная, т.е., степень ее числителя меньше степени знаменателя. Из всех рассмотренных вариантов переоборудования интегратора этому условию удовлетворяет метод Эйлера, который мы и будем использовать в работе. При попытке применить метод обратных разностей или преобразование Тастина возникает алгебраический цикл, потому что степени числителя и знаменателя передаточной функции $D_I(z)$ равны.

Описание системы

В работе рассматривается система управления судном по курсу. Ее структурная схема показана на рисунке.



Структурная схема системы стабилизации судна на курсе

Линейная математическая модель, описывающая рыскание судна, имеет вид

$$\begin{aligned} \dot{\varphi} &= \omega_y \\ \dot{\omega}_y &= -\frac{1}{T_s} \omega_y + \frac{K}{T_s} \delta \end{aligned}$$

где φ – угол рыскания (угол отклонения от заданного курса), ω_y – угловая скорость вращения вокруг вертикальной оси, δ – угол поворота вертикального руля относительно положения равновесия, T_s – постоянная времени, K – постоянный коэффициент, имеющий

размерность $rad/сек$. Передаточная функция от угла поворота руля к углу рыскания запишется в виде

$$P(s) = \frac{K}{s(T_s s + 1)}.$$

Линейная модель привода (рулевой машины) представляет собой интегрирующее звено с передаточной функцией

$$R_0(s) = \frac{1}{T_R s},$$

охваченное единичной отрицательной обратной связью. На угол перекладки руля и скорость перекладки накладываются нелинейные ограничения

$$|\dot{\delta}(t)| < 3 \text{ } ^\circ/сек, \quad |\delta(t)| < 30^\circ.$$

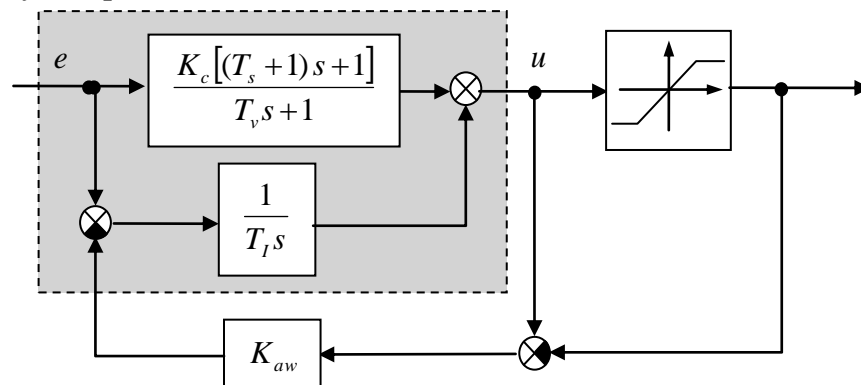
Для измерения угла рыскания используется гироскоп, математическая модель которого записывается в виде аperiodического звена первого порядка с передаточной функцией

$$H(s) = \frac{1}{T_{oc} s + 1},$$

В непрерывной системе в качестве управляющего устройства используется ПИД-регулятор с передаточной функцией

$$C(s) = K_c \left(1 + \frac{T_s s}{T_v s + 1} \right) + \frac{1}{T_I s}, \text{ где } T_v = 1 \text{ сек и } T_I = 200 \text{ сек.}$$

Для компенсации эффекта насыщения, вызванного ограниченным углом перекладки руля, используется схема с внутренней нелинейной обратной связью, охватывающей интегратор в составе регулятора.



Для реализации регулятора используется цифровой компьютер с интервалом квантования T . Регулятор в расчетной схеме заменяется не последовательное соединение трех звеньев:

4) **импульсного элемента**, который выбирает из непрерывного сигнала $e(t)$ значения $e_k = e(kT)$ в моменты квантования $t_k = kT$ (при целых k); импульсный элемент моделирует аналого-цифровой преобразователь (АЦП);

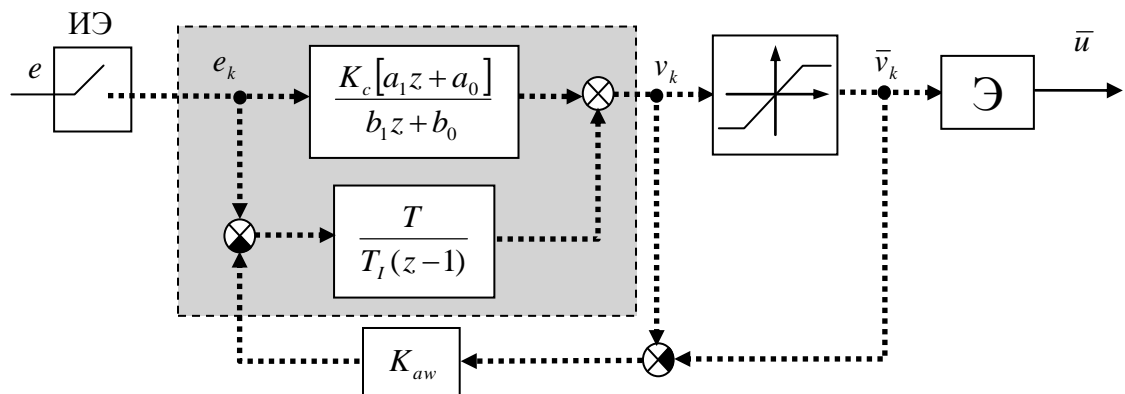
5) линейного **цифрового фильтра**, который преобразует дискретную последовательность e_k в управляющую последовательность v_k ; передаточная функция этого фильтра определяет закон управления;

6) восстанавливающее устройство (**экстраполятор**), которое восстанавливает непрерывный сигнал управления $u(t)$ из последовательности v_k ; экстраполятор моделирует

цифро-аналоговый преобразователь (ЦАП), чаще всего используется фиксатор нулевого порядка, который удерживает постоянное значение $u(t)$ в течение очередного интервала квантования:

$$u(kT + \varepsilon) = v_k, \quad 0 \leq \varepsilon < T.$$

Схема цифрового регулятора показана на рисунке:



Блок **ИЭ** обозначает импульсный элемент (АЦП), блок **Э** – экстраполятор (ЦАП). Точечные линии обозначают дискретные сигналы, сплошные линии – непрерывные.

Цифровые регуляторы обладают многими преимуществами в сравнении с классическими (непрерывными):

- отсутствует дрейф параметров элементов
- в цифровой форме можно реализовать сложные законы управления
- цифровые регуляторы легко перестраивать, настройка сводится к просто к замене алгоритма обработки измеряемых сигналов

В то же время между моментами квантования (моментами съема измеряемых сигналов и выдачи нового управляющего воздействия) система ведет себя как разомкнутая (неуправляемая). Это может привести к потере устойчивости (при больших интервалах квантования в сравнении с постоянной времени объекта) и скрытым колебаниям (колебаниям непрерывного сигнала, которые не проявляются в моменты квантования).

Для построения дискретной модели ПД-регулятора используется преобразование Тастина

$$s \leftarrow \frac{2}{T} \frac{z-1}{z+1},$$

соответствующее интегрированию по методу трапеций. Для рассматриваемого ПД-регулятора такая замена дает

$$D_{pd}(z) = \frac{K_c [a_1 z + a_0]}{b_1 z + b_0},$$

где коэффициенты равны


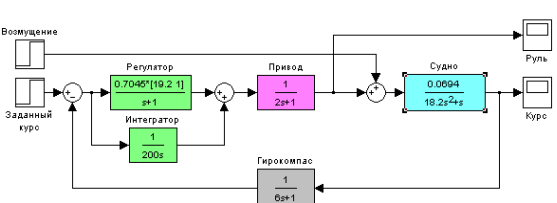
$$a_1 = T + 2(T_s + T_v), \quad a_0 = T - 2(T_s + T_v), \quad b_1 = T + 2T_v, \quad b_0 = T - 2T_v.$$


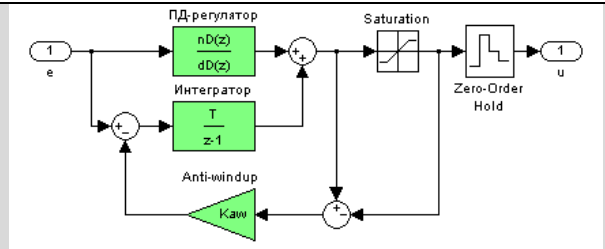
Для построения дискретной передаточной функции интегрального канала применяется метод интегрирования Эйлера (метод прямоугольников), т.е., замена

$$s \leftarrow \frac{z-1}{T}.$$

В ходе выполнения работы требуется построить цифровые реализации регулятора при различных интервалах квантования и сравнить переходные процессы в непрерывной и цифровой системе управления.

Инструкция по выполнению работы

Этап выполнения задания	Команды MATLAB
1. Сделайте свою папку рабочей папкой MATLAB.	ЛКМ по кнопке  справа от поля Current Directory
2. Создайте новый M-файл, в первой строке наберите имя скрипта sysdata для загрузки исходных данных. Добавьте в него строчку, которая задает полученное в предыдущей работе значение $K_{ав}$. Сохраните файл с именем lab7go.m и выполните его.	File – New – M-file sysdata; Kaw = ... File – Save
3. Добавьте в скрипт следующие строки (без номеров) 1 T = 1; 2 Cpd = tf(Kc*[Ts+1 1], [1 1]); 3 Dpd = c2d (Cpd, T, 'tustin'); 4 [nD,dD] = tfdata (Dpd, 'v'); Комментарий: 1 – установить значение интервала квантования 1 с; 2 – построение передаточной функции ПД-регулятора; 3 – применение преобразования Тастина к передаточной функции Cpd; в результате будет построения соответствующая дискретная передаточная функция; 4 – числитель и знаменатель дискретной передаточной функции Dpd записываются в переменные nD и dD.	
4. Выполните скрипт lab7go.m .	клавиша F5
5. Откройте окно рабочей папки.	View – Current directory
6. Соберите модель системы	 <p>Где линейная модель</p>  <p>Нелинейная модель</p> 

7. Сохраните модель в своей папке под именем lab7.mdl .	 File – Save as ...
8. Откройте подсистему Цифровая система - Регулятор . Удалите блоки ПД-регулятор и Интегратор .	ЛКМ выделить ЛКМ, нажать Delete
9. Откройте окно библиотеки блоков и скопируйте на освободившиеся места два блока Discrete Transfer Fcn (дискретная передаточная функция) из группы Discrete .	View – Library browser перетащить ЛКМ
10. Назовите новые блоки ПД-регулятор и Интегратор и задайте их параметры.	ПД-регулятор Numerator: nD Denominator: dD Sample time: T Интегратор Numerator: $[T]$ Denominator: $TI*[1 -1]$ Sample time: T
11. Добавьте на выход регулятора блок Zero-order hold (фиксатор нулевого порядка) из группы Discrete и установите интервал квантования T .	перетащить ЛКМ из окна Library browser Sample time: T
12. Соберите схему, показанную на рисунке. Скопируйте ее в отчет.	
13. Установите время моделирования 250 с.	Simulation – Simulation parameters – Stop time
14. Выполните моделирование и сравните переходные процессы в непрерывной и цифровой системах (они должны практически совпадать).	двойной щелчок ЛКМ по блокам Курс и Руль
15. Далее мы сравним переходные процессы в цифровых системах при различных интервалах квантования. Добавьте в скрипт следующие строки (без номеров): 1 close all; 2 figure(1); 3 subplot(2,1,1); 4 set(gca,'FontSize',16); 5 subplot(2,1,2); 6 set(gca,'FontSize',16); Комментарий: 1 – закрыть все рисунки; 2 – создать рисунок номер 1;	

3 – разбить его на две части по вертикали (курс и угол перекадки руля) и активизировать первый график;
4 – установить размер шрифта 16 пунктов;
5, 6 – те же операции для второго графика.

16. В цикле будем изменять интервал квантования T , строить новую дискретную модель ПД-регулятора, проводить моделирование и строить очередной график, не стирая существующий. Добавьте в скрипт следующие строки (без номеров):

```
1 aT = [2 3 5];
2 col = 'bgr';
3 for i=1:length(aT)
4   T = aT(i);
5   Dpd = c2d ( Cpd, T, 'tustin' );
6   [nD,dD] = tfdata ( Dpd, 'v' );
7   sim('lab7')
8   subplot(2,1,1);
9   plot(phi(:,1),phi(:,3),col(i));
10  hold on;
11  subplot(2,1,2);
12  plot(delta(:,1),delta(:,3),col(i));
13  hold on;
14 end;
```

Комментарий:

1 – создается массив интервалов квантования;
2 – создается символьный массив с обозначениями цветов графиков ('b' – синий, 'g' – зеленый, 'r' – красный);
3 – заголовок цикла, i – номер графика;
4 – выбрать значение интервала квантования из массива;
5 – построить дискретную модель ПД-регулятора с помощью преобразования Тастина;
6 – получить числитель и знаменатель этой модели;
7 – выполнить моделирование;
8 – перейти к первому графику;
9 – построить изменение угла курса, цвет линии определяется символом из массива col;
10 – при выводе следующего графика не стирать существующие кривые;
11-13 – те же операции для второго графика (угол перекадки руля);
14 – конец цикла

17. Остается построить (для сравнения) переходные процессы в непрерывной системе (второй столбец в массивах phi и delta). Добавьте в скрипт следующие строки (без номеров):

```
1 subplot(2,1,1);
2 plot(phi(:,1),phi(:,2),'k--');
3 legend('T=2', 'T=3', 'T=5', 'Непрерывная система');
4 h = get(gca, 'Children');
5 for i=1:4
6   set(h(i), 'LineWidth', 1.5);
```

7 end;	
Комментарий:	
1 – перейти к первому графику;	
2 – строится переходный процесс в непрерывной системе, штриховая линия ('--') черного цвета ('k');	
3 – выводится легенда;	
4 – получить массив указателей на линии;	
5-7 – цикл для изменения толщины линий	
6 – установить толщину линий 1,5;	
18. Добавьте самостоятельно аналогичные команды для второго графика (используйте второй столбец массива delta).	
19. Выполните скрипт.	клавиша F5
20. Скопируйте график в отчет.	print –dmeta ПКМ - Вставить
21. Сделайте выводы о максимальном допустимом интервале квантования, при котором переходные процессы в цифровой системе практически совпадают с переходными процессами в непрерывной системе.	

Таблица коэффициентов

Вариант	T_s , сек	K , рад/сек	T_R , сек	T_{oc} , сек
1.	16.0	0.06	1	1
2.	16.2	0.07	2	2
3.	16.4	0.08	1	3
4.	16.6	0.07	2	4
5.	16.8	0.06	1	5
6.	17.0	0.07	2	6
7.	17.2	0.08	1	1
8.	17.4	0.07	2	2
9.	17.6	0.06	1	3
10.	17.8	0.07	2	4
11.	18.0	0.08	1	5
12.	18.2	0.09	2	6
13.	18.4	0.10	1	1
14.	18.6	0.09	2	2
15.	18.8	0.08	1	3
16.	19.0	0.07	2	4
17.	19.2	0.08	1	5
18.	19.4	0.09	2	6
19.	19.6	0.10	1	1
20.	18.2	0.0694	2	6

Литература

1. Джонс Э., Саттон Д. Office 97 Библия пользователя. - К.-М.: "Диалектика", 1997.
2. Пасько В. Access 97 (Русифицированная версия) - К: Изд.группа ВНУ, 1998.
3. Информатика. Технологии работы на компьютере/ под ред. Н.В.Макаровой. М.: Финансы и статистика. 2003. 865 с
4. Мартин Дж. Организация баз данных вычислительных системах.-М.:Мир.1980. - 664 с
5. Костин А.Е., Шаньгин Б.Ф. Организация и обработка структур данных в вычислительных системах. Учебн.пособие для вузов . - М.: Высш. шк. 1987. - 248 с.
6. Microsoft Access: наглядно и конкретно. М:Персанг. 1997 г
7. Гайдаманил Н. А. Автоматизирования информационные системы, базы данных. Вводный курс. М.:2002, 368с
8. Информатика. Практикум по технологии работы на компьютере//под ред. Н.В.Макаровой. М.: Финансы и статистика. 2003. -256 с.
9. Крамм Р. Системы управления базами данных dBASEII и dBASEIII для персональных компьютеров. - М.: Финансы и статистика. 1988.-283 с.
10. Фигурнов В. Э. IBM PC для пользователя. М-ИНФРА. 2001, 480 с
11. <http://www.infocity.uk>
12. <http://www.delphi.ru>
13. <http://www.delphimaster.ru>
14. <http://www.leader.ru>