**Лабораторная работа 4**

**Автоматизация расчета модальных регуляторов для одномерных непрерывных и цифровых систем управления**

***Цель работы:*** освоить работу с программами автоматизации расчета модальных регуляторов для одномерных непрерывных и цифровых систем управления с постоянными параметрами с желаемым характеристическим полиномом.

**4.1. Основные сведения**

В интегрированной системе MATLAB/Simulink имеются средства автоматизации проектирования различных типов регуляторов, с некоторыми из которых вы познакомились при выполнении предыдущих работ. В то же время синтезу систем с модальным управлением уделено недостаточно внимания, несмотря на то, что такие системы широко используются на практике. Желание исправить этот недостаток побудило разработать программы, предназначенные для автоматизации синтеза систем с модальным управлением на основе заданного распределение корней характеристического уравнения замкнутой САУ. Распределение корней в свою очередь полностью определяет движение (динамику) системы.

Исходя из требований к переходному процессу, определяется желаемый характеристический полином и желаемое расположение полюсов замкнутой системы, например, с использованием метода стандартных полиномов. *Примечание:*

В технической литературе приводятся различные наборы стандартных характеристических полиномов 1-8 порядков и соответствующие им графики переходных процессов с указанными на них показателями качества. Наиболее часто в практике проектирования используются следующие полиномы:

* полином Ньютона;
* полином Баттерворта;
* полином Бесселя.

Стандартные полиномы содержат нормирующий множитель ω0, называемый среднегеометрическим корнем. Его значение изначально равно ω0=-1. Это значение определяет скорость протекания стандартного переходного процесса tст. Значение желаемого времени переходного процесса, заданное из условий проектирования, называемое tж, определяет требуемое значение среднегеометрического корня. Это значение находится из соотношения ω0=tст/tж. Стандартный полином, например, полином Баттерворта 4-го порядка p4 + 2,613ω0 p3+ 3,414ω02 p2+ 2,613ω03 p+ ω04, нормируется вычисленным значением среднегеометрического корня. После нормировки мы получаем желаемый характеристический полином, который обеспечит требую динамику замкнутой системы.

Для расчёта аналоговых фильтров-прототипов Баттерворта и Бесселя в программе MATLAB используются стандартные функции buttap и besselap:

>> [z, p, k] = buttap(n);

>> [z, p, k] = besselap(n);

Для вычисления стандартного полинома Ньютона, все корни которого кратные, можно использовать команды:

>> n= input('Введите порядок системы n = ');

>> r(1:n)=-1;

>> p=poly(r);

Применяя метод стандартных полиномов, или другими словами метод заданного расположения полюсов, можно вычислить матрицу коэффициентов обратных связей, ко­торая обеспечит *любое* желаемое расположение этих полюсов на комплекс­ной плоскости при условии, что система является *управляемой*.

Последовательность проектирования здесь следующая:

1. Проверяется управляемость пары матриц {A, B}, например

>>A = […];

>>B = […];

Co = ctrb (A, B)

unco = length (A) − rank (Co) ;% Число неуправляемых мод

if unco == 0

disp ( 'Система полностью управляема' )

else

T = 'Число неуправляемых мод равняется ';

disp ([T unco])

end

1. Исходя из требований к переходному процессу, определяется желаемый характеристический полином и вектор корней характеристического уравнения, то есть вектор p − желаемое расположение полюсов замкнутой системы.
2. Вычисляется матрица коэффициентов обратных связей K, обеспечивающая заданное расположение полюсов в проектируемой системе. При этом используют следующие функции:

>>K = acker(A,B,p) % Для одномерной системы

>> K = place(A,B,p) % Для одномерных и многомерных систем

Здесь аргументы A и B – матрицы ss-модели, p – вектор желаемых полюсов. Возвращаемая величина K – матрица коэффициентов обратных связей.

Функция **acker** предназначена для расчета одномерных систем. Функция **place** может быть приме­нена как для одномерных, так и для многомерных систем и использует специальный алгоритм, который гарантирует высокую точность. При распределении корней по Ньютону следует использовать функцию **acker**.

Все описанные выше процедуры касались расчета непрерывных систем с модальным управлением. Для синтеза цифровых систем на основе аналоговых прототипов используются рассмотренные ранее методы дискретизации, то есть получение дискретных матриц Ad, Bd, Cd, Dd объекта управления, и вектора желаемых полюсов дискретной системы *zi=eSiT*, где *si* − желаемые полюса непрерывной системы, T − период дискретизации.

*Примечание*: для вычисления полюсов непрерывной системы, если известны полюса дискретной системы может использоваться выражение

*si =* 1*/T·*ln *zi*.

Рассмотренная последовательность операций может использоваться как с непрерывным, так и с дискретным описанием проектируемых систем и легко автоматизируется.

**4.2. Синтез модального регулятора с использованием метода**

**Фаддеева-Леверье**

Положим, что объект управления задан уравнением в пространстве состояний

|  |  |
| --- | --- |
| , | (1) |

где *x* – вектор состояния системы, *u* – сигнал управления, *А* – системная матрица, *В* – матрица управления, *n* – порядок системы.

Модальный регулятор в виде обратной связи по вектору состояния реализует закон управления следующего вида:

|  |  |
| --- | --- |
| , | (2) |

где *R*ос– вектор коэффициентов модального регулятора в цепи обратной связи, *R*п– коэффициент передачи модального регулятора в прямой цепи, *g*(*k*) – задающее воздействие.

Для систем с одним входом задача нахождения матрицы *R*ос имеет единственное решение и существует ряд известных алгоритмов ее нахождения: по формуле Аккермана в системе МАТЛАБ реализована подпрограмма **acker**, а по алгоритму Каутского-Николса-Ван Дурена подпрограмма **place**.

С учетом уравнения закона управления (2) запишем уравнение (1) в виде:

|  |  |
| --- | --- |
| . | (3) |

Структура замкнутой системы с модальным регулятором показана на рис. 4.1.

|  |
| --- |
|  |
| Рис. 4.1. Структура системы управления с модальным регулятором |

Характеристический полином системы (1) задается выражением

,

где коэффициенты характеристического полинома *fi* (*i* = 1,…*n*) могут быть определены с помощью алгоритма Фаддеева-Леверье

 (4)

где *Q*0 = *I*, *I* – единичная nn матрица, *R* – вспомогательная nn матрица,  – след матрицы R.

Рассмотрение процедуры расчета модального регулятора и метода Фаддеева-Леверье позволяет разработать универсальный, пригодный для непрерывных и цифровых систем автоматического управления метод расчета параметров модального регулятора, основанный на рассматриваемой ниже модификации алгоритма Фаддеева-Леверье.

Передаточная матрица объекта управления по вектору состояния может быть определена как

,

где 

Аналогичная передаточная матрица замкнутой системы с модальным регулятором

.

Таким образом, коэффициенты характеристического полинома замкнутой системы можно выразить через коэффициенты характеристического полинома разомкнутой системы *F*(*z*) и элементы произведения матриц *R*ос*Q*(*z*)*B*

, (5)

где *Qk* – матрицы, получаемые с помощью алгоритма (4).

Положим далее, что желаемый характеристический полином системы (3), обеспечивающий проектируемой системе требуемые динамические характеристики имеет вид

.

Естественно, что коэффициенты характеристических полиномов желаемой и замкнутой систем должны совпадать, поэтому

, (6)

или при введении невязки

.

Для всех *k* (*k* = 1… *n*) выражение для невязки запишется в матричном виде как



где .

Тогда вектор *R*ос модальных связей может быть определен из решения алгебраического уравнения

.

В соответствии с описанным алгоритмом была разработана функция расчёта коэффициентов модальных регуляторов в среде MATLAB, которая далее будет использована в программе автоматизированного проектирования систем с модальным управлением. Текст функции представлен на рис. 4.2.

function Roc = fad\_lev(A, B, p)

n = length(A);

fg = poly(p); % Коэффициенты желаемого ХП

I = eye(n);

[R{1: n}] = deal(A);

[Q{1: n}] = deal(I);

f = ones(1, n+1);

for k = 2: n+1

f(k) = -trace(R{k-1})/(k-1);

Q{k} = (R{k-1}+f(k)\*I);

R{k} = A\*Q{k};

end

S=[ ]; E=[ ];

for j = 2: n+1

S = [S Q{1, j-1}\*B];

E = [E fg(j) - f(j)];

end

Roc = E\*inv(S);

end

Рис 4.2. Функция расчета коэффициентов модального регулятора

Функция принимает следующие исходные данные непрерывной или цифровой системы: матрицы A и B, p − вектор-строку желаемых корней характеристического уравнения. В результате, функция возвращает вектор-столбец коэффициентов модального регулятора.

Представленная функция **fad\_lev** может использоваться в задачах проектирования одномерных систем с модальным управлением также, как и стандартные функции **acker** и **place**.

**4.3. Программы автоматизация расчета модальных регуляторов**

Для рассмотренных алгоритмов расчёта коэффициентов модальных регуляторов в программе App Designer среды MATLAB были разработаны две программы, графический интерфейс которых, представлен на рис. 4.3.

Обе программы имеют одинаковые интерфейсы и функциональные возможности. Отличие заключается в используемых функциях расчета модальных регуляторов. Первая программа использует стандартные функции **acker** и **place**, а вторая функцию **fad\_lev**, описанную выше.

После запуска программы, пользователь вводит исходные данные непрерывной системы (порядок системы, количество выходов, значения матриц A, B и C), выбирает желаемый характеристический полином из группы стандартных (Ньютона, Баттерворта или Бесселя) и требуемое время переходного процесса, а на выходе получает, с помощью нажатия соответствующей кнопки, либо коэффициенты модального регулятора непрерывной, либо дискретной системы с выбранным периодом дискретности. Выбор типа системы (непрерывная или дискретная) осуществляется независимо. Программа защищена от ошибочного, либо неполного ввода исходных данных. Ошибки вызывают появление окон диагностических сообщений.

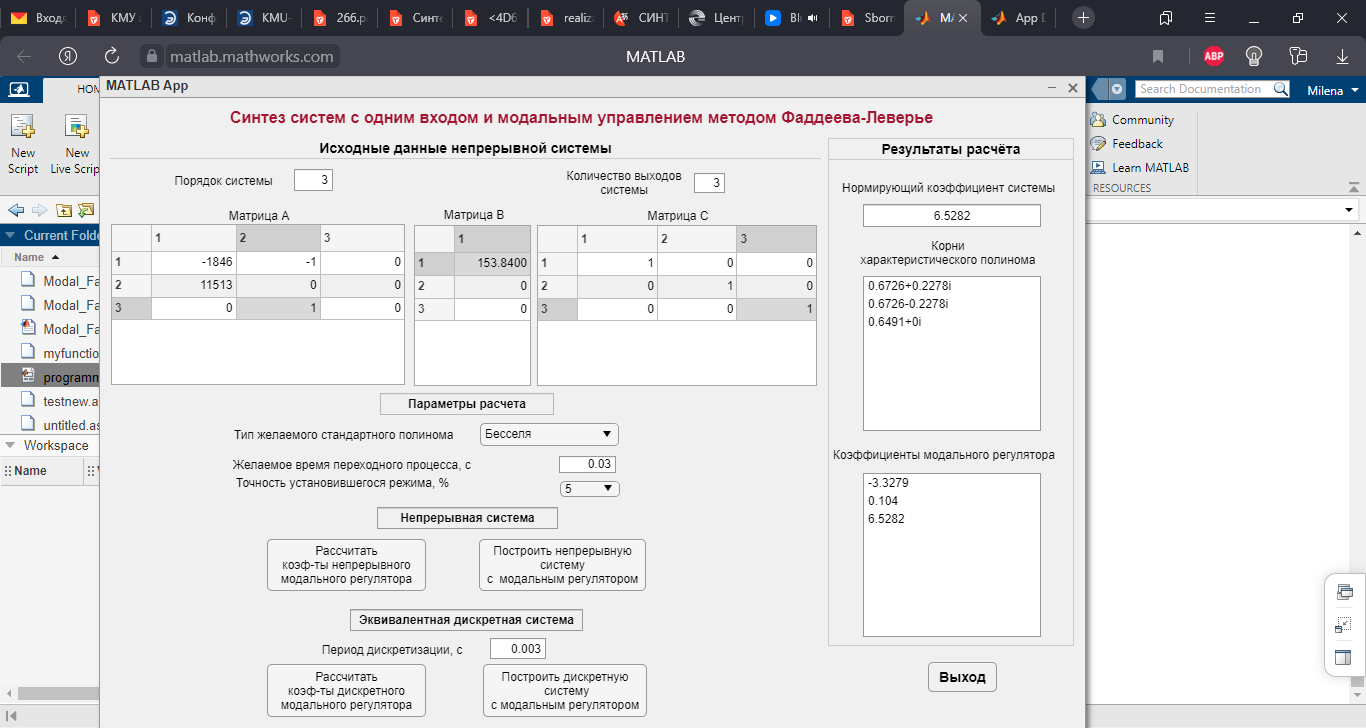


Рис. 4.3. Графический интерфейс программы

Программа позволяет не только наблюдать результаты расчёта в окне результатов, но и выводить их в командное окно среды MATLAB в виде текста отчета. Также реализована возможность построения Simulink-моделей систем с модальным управлением.

Simulink-модели систем реализованы программно в соответствии с векторно-матричной структурой, показанной на рис. 4.1. После запуска на симуляцию этих моделей можно увидеть переходные процессы как по выходной переменной, так и по всем переменным состояния. В дальнейшем эти модели можно использовать для различных задач проектирования.

**4.4. Порядок выполнения работы**

1. Изучить методику расчета коэффициентов модального регулятора с использованием стандартных средств проектирования системы MATLAB.

2. Изучить метод синтеза модального регулятора на основе модифицированного метода Фаддеева-Леверье.

3. Используя описанные программы автоматизации (файлы modal.mlapp и modal\_Fad\_Lev.mlapp), произвести расчет непрерывной и дискретной следящих систем с модальным управлением. Передаточную функцию непрерывного объекта управления (двигатель постоянного тока независимого возбуждения) выбрать в соответствии с вариантом задания из табл. 1 (лекция 2). В качестве желаемого полинома выбрать любой из стандартных. Желаемое время переходного процесса и период дискретизации выбрать такими же, какими вы выбирали их ранее (при выполнении лаб. работы 2).

4. Представить в отчете графический интерфейс с исходными данными по вашему варианту и результаты расчета, схемы simulink-моделей и графики переходных процессов.

5. Сравнить результаты моделирования с результатами, полученными при выполнении лабораторной работы 2.

6. Ознакомиться с текстом программного кода программ.

7. На основании полученных результатов моделирования сделать выводы и оценку удобства использования предложенных программ для автоматизации расчетов.