**МИНОБРНАУКИ РОССИИ**

**Санкт-Петербургский государственный**

**электротехнический университет**

**«ЛЭТИ» им. В.И. Ульянова (Ленина)**

**Кафедра САУ**

отчет

**по Индивидуальному домашнему заданию**

**по дисциплине «Современные методы теории управления»**

Тема: СИНТЕЗ РОБАСТНОГО РЕГУЛЯТОРА МЕТОДОМ H∞

Вариант 19

|  |  |  |
| --- | --- | --- |
| Студент гр. 7492 |  | Бондарчук А.П. |
| Преподаватель |  | Скакун А.Д. |

Санкт-Петербург

2021

# 1. Расчет нормы Харди для объекта управления

Передаточная функция для объекта управления имеет вид (1.1).

|  |  |
| --- | --- |
|  | (1.1) |

Норма Харди для передаточной функции вычисляется как (1.2).

|  |  |
| --- | --- |
|  | (1.2) |

где – передаточная функция системы, – супремум частотной передаточной функции или точная верхняя граница частотной ПФ.

Модуль ЧПФ вычисляется как (1.3).

|  |  |
| --- | --- |
|  | (1.3) |

Для вычисления точной верхней границы вычисляется производная модуля ЧПФ и находится наибольшее значение ЧПФ в точках экстремума.

Для нахождения нормы Харди был использован следующий скрипт:

clc,clear,close

syms s w

H\_s=(s+8)/((s+10)\*(s^2+4\*s+100));

H\_jw=subs(H\_s,s,j\*w);

H\_inv\_jw=subs(H\_s,s,-j\*w);

H\_mod=simplify(H\_jw\*H\_inv\_jw);

H\_mod\_diff=simplify(diff(H\_mod,w));

rez=vpasolve(H\_mod\_diff,w)

for i=1:length(rez)

if(isreal(rez(i)))

norm(i)=eval(subs(H\_mod,w,rez(i)));

end

end

В результате получим модуль ЧПФ (1.4).

|  |  |
| --- | --- |
|  | (1.4) |

Результат работы программы изображен на рисунке 1.1.

ЧПФ имеет экстремумы в точках -9.63,0,9.63.

Норма Харди равняется 0.0005293.

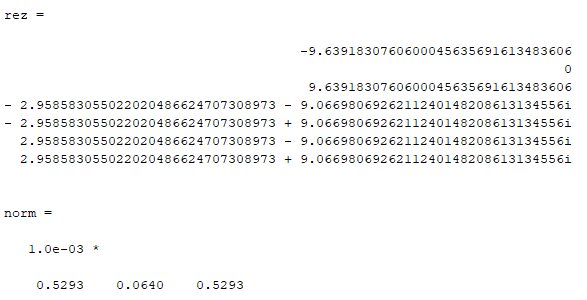


Рисунок 1.1 – Результат вычисления нормы Харди

Ответ:

# 2 Нахождение весовых функций

Структурная схема системы представлена на рисунке 2.1.

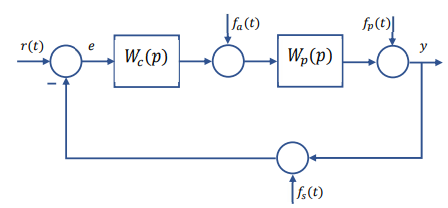


Рисунок 2.1 – Структурная схема исходной системы

Для системы заданы следующие воздействия:

.

Требуется обеспечить:

## 2.1 Расчет функции чувствительности S

1. Для обеспечения нулевой ошибки по задающему воздействию система должна иметь первый порядок астатизма, следовательно ;

2. Определим ограничения функции чувствительности в зависимости от постоянного возмещения . По теореме о конечном значении получим (2.1).

|  |  |
| --- | --- |
|  | ( 2.1) |

3. Для обеспечение требования значение ЛАХ на частоте должно составлять -20 Дб. Так как между и меньше 1-й декады сдвинем на частоту 1 рад/с и выберем наклон ЛАХ функции чувствительности S равный +20 дб/дек.

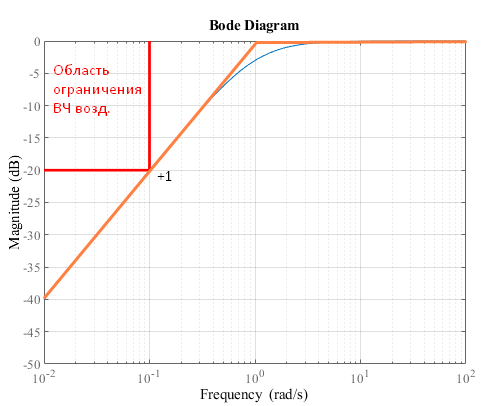


Рисунок 2.1 – ЛАХ функции чувствительность S

ПФ функции чувствительности зададим как (2.2).

|  |  |
| --- | --- |
|  | (2.2) |

## 2.2 Расчет функции дополнительной чувствительности T

В данном случае функция дополнительной чувствительность T определяется исходя из требований к времени регулирования *,* следовательно, частота среза .

Так как частота 0.6 рад/с меньше частоты среза функции S(s), то для функции дополнительной чувствительность T(s) выберем частоту среза .

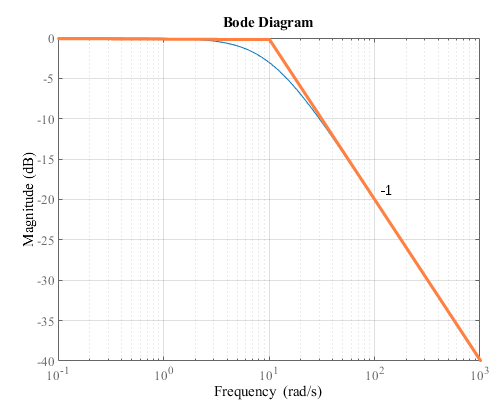


Рисунок 2.2 – ЛАХ функции дополнительной чувствительности T

Функция дополнительной чувствительности T будет иметь вид (2.3).

|  |  |
| --- | --- |
|  | (2.3) |

## 2.3 Весовые функции

Весовые функции вычисляются по формулам (2.4):

|  |  |
| --- | --- |
|  | (2.4) |

# 3. Синтез и моделирование робастного регуляторы

ПФ объекта управления имеет вид (1.1). Для данного объекта заданы следующие возможные отклонения:

## 3.1 Исследование возможных отклонений

Запишем ПФ неопределённости, как (3.1).

|  |  |
| --- | --- |
|  | (3.1) |

где – передаточная функция реального объекта (с отклонениями параметров), – передаточная функция реального объекта.

ЛАХ неопределенностей должны лежать ниже весовой функции . Если это условие не выполнено, то весовую функцию необходимо изменить.

Построим ЛАХ неопределённости при различных отклонениях a1 и a3.

b1=1;

b2=8;

a1=10;

a2=4;

a3=100;

W2=tf([0.1 1],[1]);

Wp\_num=[b1 b2];

Wp\_den=conv([1 a1],[1 a2 a3]);

Wpn=tf(Wp\_num,Wp\_den);

figure

set(0,'DefaultAxesFontSize',14,'DefaultAxesFontName','Times New Roman');

set(gcf,'Color',[1 1 1]);

bodemag(W2);

hold on;

grid on;

leg=["W2(s)"];

for i=-25:10:15

for j=-20:10:20

Wp\_den=conv([1 a1\*(i/100+1)],[1 a2 a3\*(j/100+1)]);

Wp=tf(Wp\_num,Wp\_den);

W\_delta=(Wp/Wpn)-1;

bodemag(W\_delta);

leg=[leg "Отклонение a1 на " + i + "%, b1 на " + j + "%"];

end

end

legend(leg);

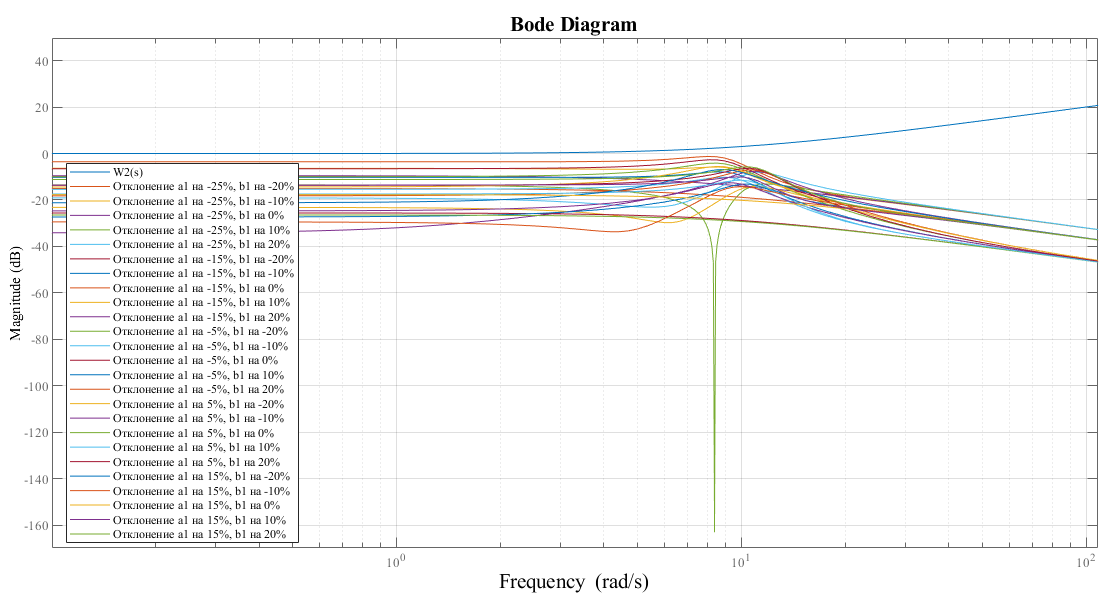


Рисунок 3.1 – ЛАХ неопределённости при различных отклонениях параметров

Как видно из рисунка 3.1 при всех возможных заданных отклонениях параметров ЛАХ неопределённости лежат ниже ЛАХ весовой функции , следовательно, ее можно применять для дальнейшего синтеза робастного регулятора.

# 3.1 Синтез робастного регулятора

Для синтеза регулятора создадим модель в Simulink.

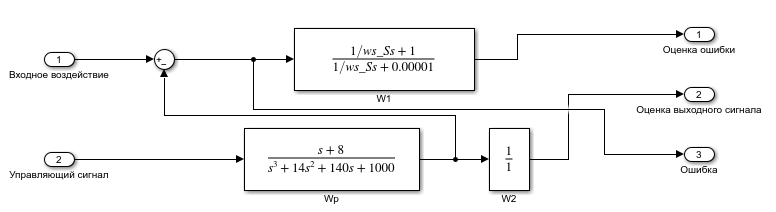


Рисунок 3.2 – Модель для синтеза регулятора

Сместим нулевой полюс функции от начала координат, тк функция синтеза не умеет работать с нулевыми полюсами. Для синтеза робастного регулятора воспользуемся функцией *hinflmi*.

clc,clear,close

Wp\_num=[b1 b2];

Wp\_den=conv([1 a1],[1 a2 a3]);

ws\_S=1;

ws\_T=10;

[A,B,C,D]=linmod('obj');

sys=ltisys(A,B,C,D);

sys =sderiv(sys,2,[1/ws\_T 1]);

[gopt,reg]=hinflmi(sys,[1 1]);

[cnum, cden]=ltitf(reg);

Результат работы программы отображен на рисунке 3.3.

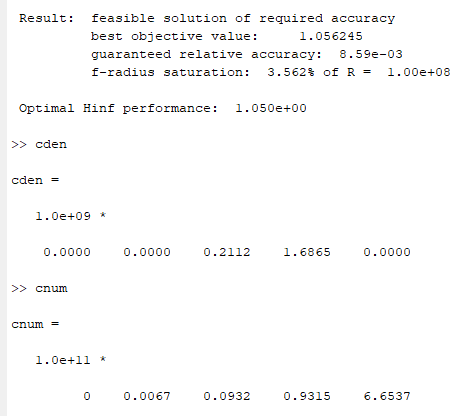


Рисунок 3.3 – Результат синтеза робастного регулятора

В результате синтеза был получен регулятор 3го порядка, показатель качества которого равен 1.050, так как он близок к единице, то результат синтеза является удовлетворительным.

## 3.3 Моделирование системы с регулятором

Для проверки корректности работы полученного регулятора была составлена модель в Simulink c учетом НЧ возмущения и постоянного возмущения в выходной переменной.

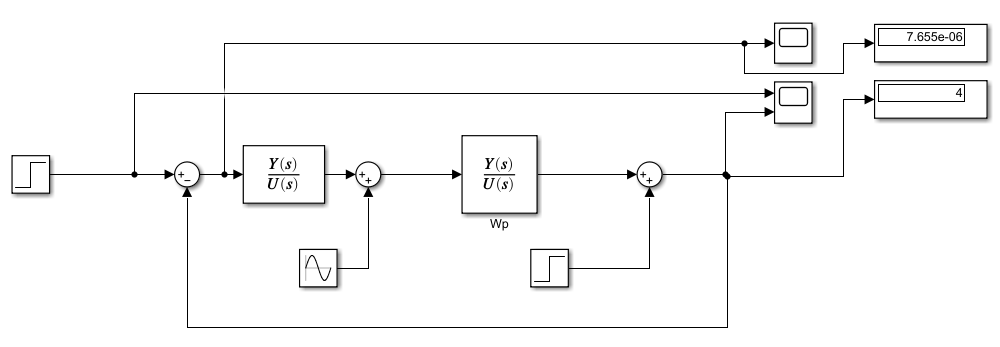


Рисунок 3.4 Система с регулятором

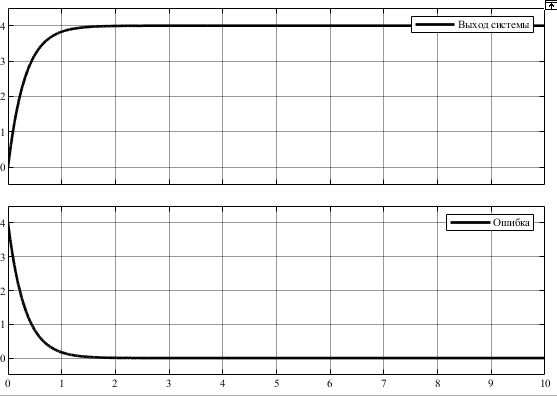


Рисунок 3.5 – Переходные процессы в системе без возмущений

Как видно из рисунка 3.5 в системе без приложенных возмущений и отклонений выполняются требования как по времени регулирования так и по перерегулированию.

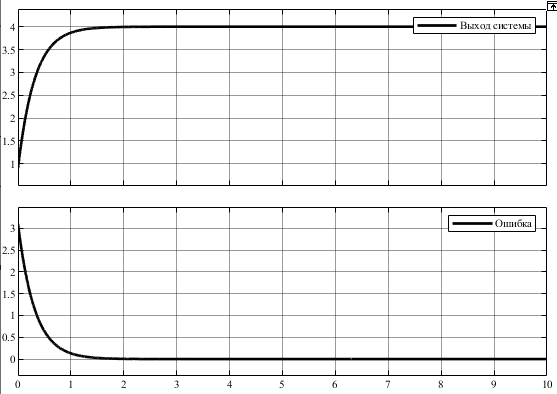


Рисунок 3.6 – Переходные процессы в системе с постоянным и низкочастотным возмущениями

Как видно из рисунка 3.6 при приложении возмущений статическая ошибка равняется нулю, также выполняются требования к быстродействию и перерегулированию.

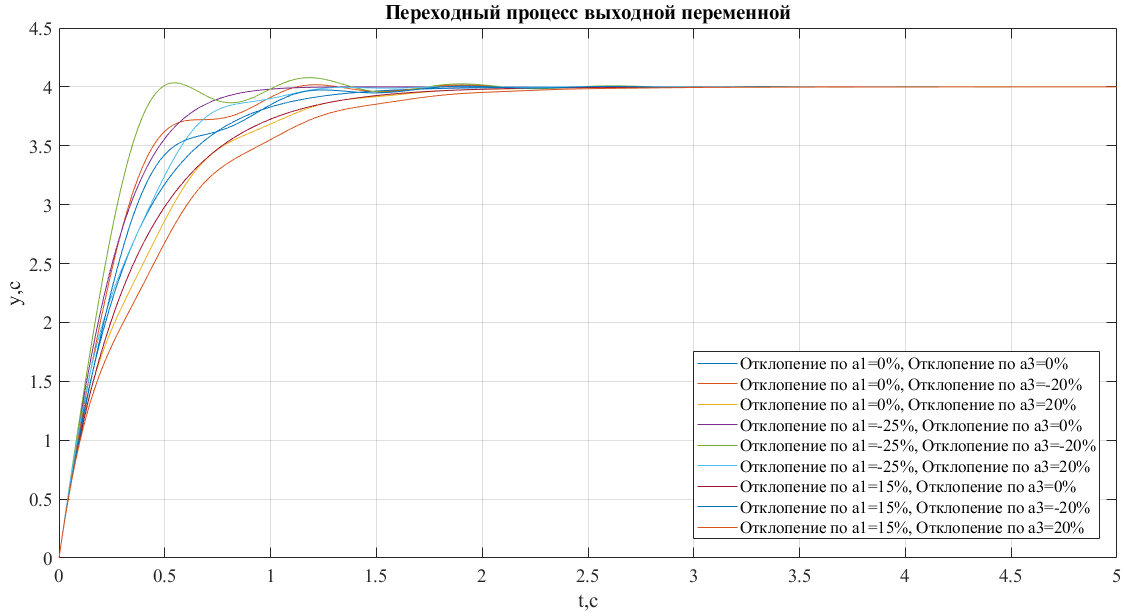


Рисунок 3.7 – Переходные процессы выходной переменной при отклонениях .

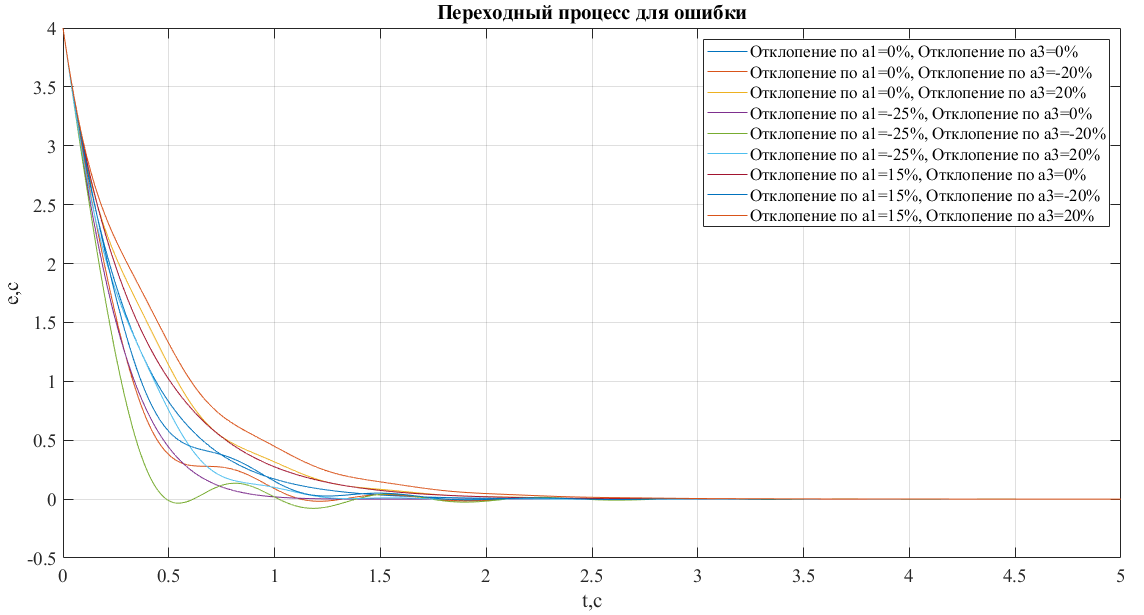


Рисунок 3.8 – Переходные процессы ошибки при отклонениях .

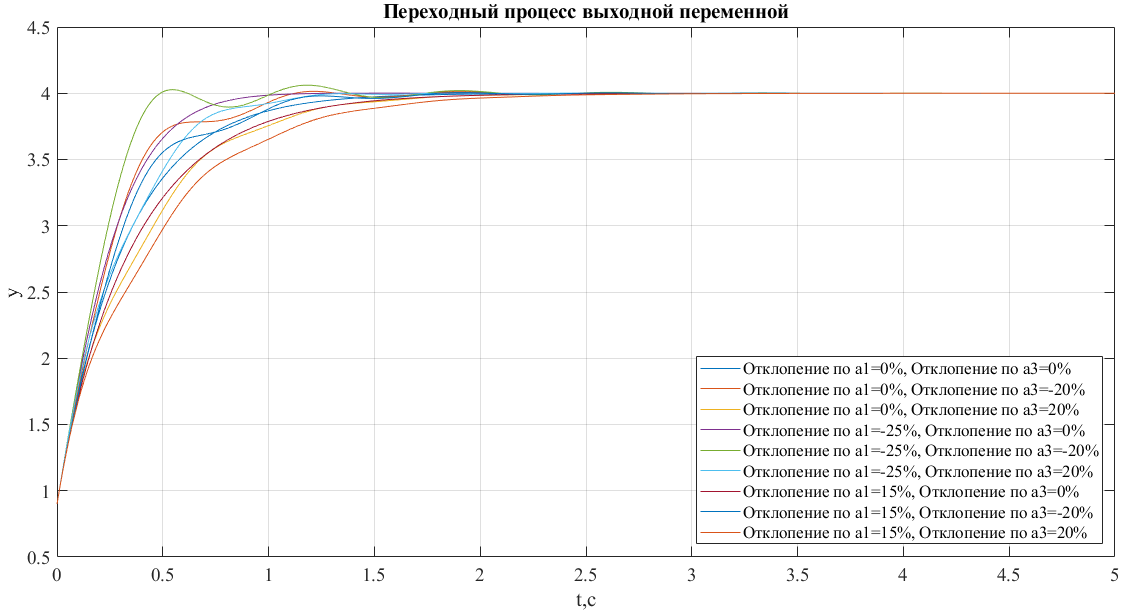


Рисунок 3.9 – Переходные процессы выходной переменной при отклонениях и наличии возмущающих воздействий

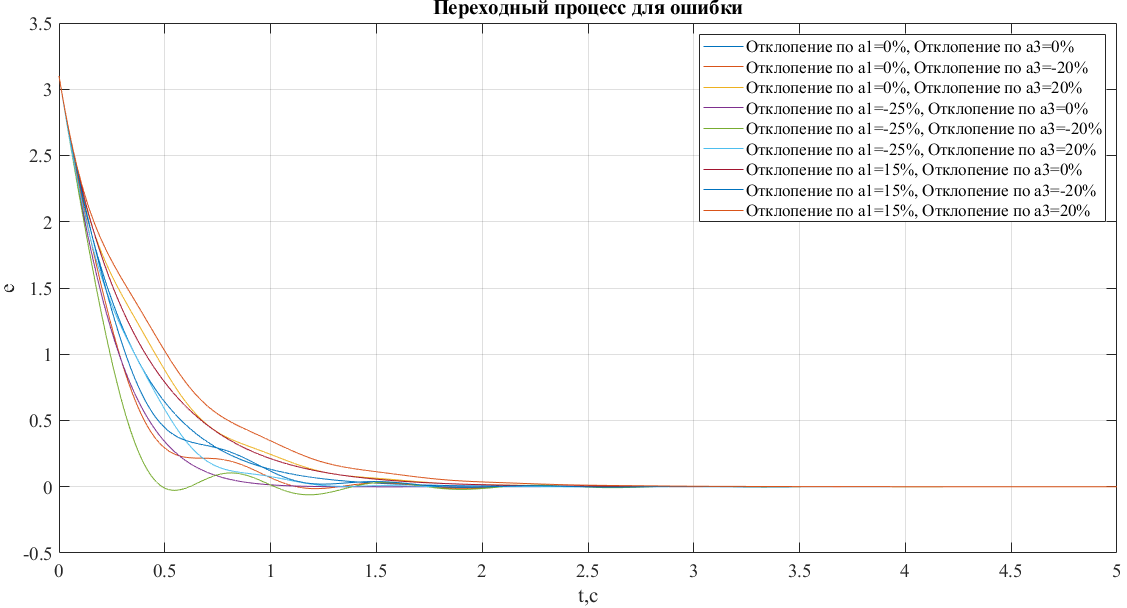


Рисунок 3.10 – Переходные процессы ошибки при отклонениях и наличии возмущающих воздействий

**Выводы:**

В результате выполнения работы был создан регулятор, который обеспечивает системе следующие характеристики:

1. Нулевая статическая ошибка по постоянному задающему и постоянному возмущающему воздействиям.

2. Малая (<0.05) ошибка по низкочастотному возмущающему воздействию.

3. Удовлетворяет требованию по быстродействию при любых заданных отклонениях параметров ОУ и наличии возмущений. Время регулирования в лучшем случае составляет 0.6с, в худшем случае 1.5с.

4. Нулевое перерегулирование при всех допустимых вариациях параметров системы кроме случая отклонения . (Перерегулирование в этом случае <1%, что является допустимым при таких отклонениях);

Согласно полученным результатам синтезированный регулятор может обеспечивать требуемые показатели качества при заданной вариации параметров ОУ и наличии возмущающих воздействий, следовательно, регулятор является робастным для данной системы.