Министерство образования Республики Беларусь

Учреждение образования «Белорусский государственный университет   
информатики и радиоэлектроники»

Факультет информационных технологий и управления

Кафедра систем управления

Дисциплина: Математические основы теории систем

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

к курсовой работе

на тему

Математические модели систем управления и методы оптимизации

БГУИР КР1-53 01 07 02 ПЗ

Студент: гр. 222403 Баранов В.В

Руководитель: старший преподаватель Стасевич Н.А.

Минск 2023

Учреждение образования

«Белорусский государственный университет информатики   
и радиоэлектроники»

Факультет информационных технологий и управления

ЗАДАНИЕ

по курсовой работе

Студенту   Баранову Владиславу Викторовичу*––*

1. Тема работы: Математические модели систем управления и методы оптимизации

2. Срок сдачи студентом законченной работы 21 декабря 2023г

3. Исходные данные к заданиям работы:

*1.*Передаточная функция исследуемой системы имеет вид:

*;*

2.Математическая модель задачи линейного программирования:

Функция цели *F(x)=4x1-4x2+4x3-x4(min); ограничения представлены системой:*

*-3x1+3x2+3x3+4x4=21;*

*-4x1+2x2-3x3-2x4>=-21;*

*-4x1+x2-4x3=<-21;*

*-3x1+2x2+4x3=<9;*

*x1,x2,x3>=0;*

3.Целевая нелинейная функция *F(x)=4x1^2+4x2^2-3x1x2-x1-7x2(max);*

*Линейные ограничения имеют вид:*

*-2x1+3x2<=0;*

*44x1+9x2-450<=0;*

*x1>=0; x2>=0.*

4. Содержание расчетно-пояснительной записки

Введение.

1.Исследование системы управления.

1.1. Описание tf-, zpk-, ss- форм. Вычисление и построение в MATLAB временных характеристик системы.

1.2.Построение в МАТLAB частотных характеристик и вид асимптотических ЛАЧХ и ЛФЧХ.

1.3.Составление уравнений состояния в нормальной и канонической формах . Результаты моделирования системы.

1.4.Решение уравнений состояния в канонической форме.

1.5.Выводы.

2.Линейное программирование.

2.1.Расчет оптимального плана и экстремального значения функции цели.

2.2.Исследование двойственной задачи линейного программирования.

2.3.Нахождение целочисленного решения задачи.

2.4.Выводы.

3.Нелинейное программирование.

3.1. Нахождение безусловного эктремума функции F(x) .

3.2.Нахождение экстремума функции F(x) c учетом системы ограничений.

3.3.Выводы.

Заключение.

5.Дата выдачи задания 8 сентября 2023г.

6.Сроки выполнения отдельных разделов работы : Раздел 1 - к 10.10.23г., раздел 2 - к 14.11.23г.,

раздел 3 - к 12.12.23г., оформление - к 20.12.23г.

РУКОВОДИТЕЛЬ

Н.А.Стасевич*.–––––––А–––*

(подпись)

Задание принял к исполнению *–––––––\_\_\_\_\_чч\_\_ЭЭЭЭЭ\_\_\_\_\_\_\_*

(дата и подпись студента)

**СОДЕРЖАНИЕ**

[Введение 5](#_Toc153411098)

[1 Исследование систем управления 6](#_Toc153411099)

[1.1 Описание tf-, zpk-, ss- форм. Вычисление и построение в MATLAB временных характеристик системы 6](#_Toc153411100)

[1.2 Построение в МАТLAB частотных характеристик и вид асимптотических ЛАЧХ и ЛФЧХ 10](#_Toc153411101)

[1.3 Составление уравнений состояния в нормальной и канонической формах. Результаты моделирования системы 12](#_Toc153411102)

[1.4 Решение уравнений состояния в канонической форме 17](#_Toc153411103)

[2 Линейное программирование 19](#_Toc153411104)

[2.1 Расчет оптимального плана и экстремального значения функции цели 19](#_Toc153411105)

[2.2 Исследование двойственной задачи линейного программирования 22](#_Toc153411106)

[2.3 Решение целочисленной задачи по методу Гомори 26](#_Toc153411107)

[3 Нелинейное программирование 29](#_Toc153411108)

[3.1 Нахождение безусловного экстремума функции *F*(*x*) 29](#_Toc153411109)

[3.2 Нахождение экстремума функции *F*(*x*) с учетом системы ограничений 33](#_Toc153411110)

[Заключение 47](#_Toc153411111)

[Литература 47](#_Toc153411112)

# Введение

Методы оптимизации находят широкое применение в различных областях науки и техники. Эти методы успешно применяются в решении задач технического проектирования устройств и систем, организационно-экономических и других задач.

В наиболее общем смысле теория оптимизации представляет собой совокупность фундаментальных математических результатов и численных методов, которые позволяют найти наилучший вариант из множества альтернатив и избежать при этом полного перебора и оценивания возможных решений. Знание методов оптимизации является необходимым для инженерной деятельности при создании новых, более эффективных и менее дорогостоящих систем, а также при разработке методов повышения качества функционирования существующих систем.

Целью курсового проекта является построение математических моделей линейных систем управления и их моделирование, а также изучение методов оптимизации задач линейного и нелинейного программирования.

Первый раздел посвящен анализу заданной с помощью передаточной функции системы. В этом разделе для этой функции построены переходные и логарифмические амплитудно- и фазочастотная характеристики, а также построены схемы модели в пространстве состояний в нормальной и канонической формах и решено уравнение состояния в канонической форме.

Второй раздел посвящен решению задач линейного программирования. В этом разделе приведено решение прямой задачи линейного программирования и соответствующей ей двойственной задачи, а также целочисленной задачи с помощью симплекс-таблиц.

Третий раздел посвящен решению задач нелинейного программирования. В этом разделе приведено решение такой задачи без ограничений методами Ньютона-Рафсона и наискорейшего спуска, а также с ограничениями методами допустимых направлений Зойтендейка, Куна-Таккера и линейных комбинаций. Результаты решения различными методами сравнены между собой.

# 1 Исследование систем управления

## **1.1 Описание tf-, zpk-, ss- форм. Вычисление и построение в MATLAB временных характеристик системы**

Передаточная функция системы – отношение изображения выходного сигнала к входному сигналу при нулевых начальных условиях.

Передаточная функция имеет вид:

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

Характеристическое уравнение системы определяется знаменателем передаточной функции и имеет вид:

(1.2)

Найдем корни характеристического уравнения:

.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |

. (1.3)

Передаточная функция в форме нулей и полюсов имеет вид:

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

Импульсная переходная характеристика – процесс изменения сигнала на выходе при подаче на вход -функции.

Определим как обратное преобразование Лапласа от передаточной функции:

. (1.5)

Разложим передаточную функцию (1.4) на сумму простых слагаемых:

|  |  |
| --- | --- |
|  | (1.6) |

Найдем коэффициенты по методу неопределенных коэффициентов:

Передаточная функция примет вид:

|  |  |
| --- | --- |
|  | (1.7) |

В соответствии с формулой (1.5), таблицами преобразования Лапласа, найдем импульсную переходную характеристику:

. (1.8)

Вид импульсной переходной характеристики, построенный в пакете Matlab, представлен на рисунке 1.1.

Переходная характеристика – процесс изменения сигнала на выходе при подаче на вход единичного ступенчатого воздействия.

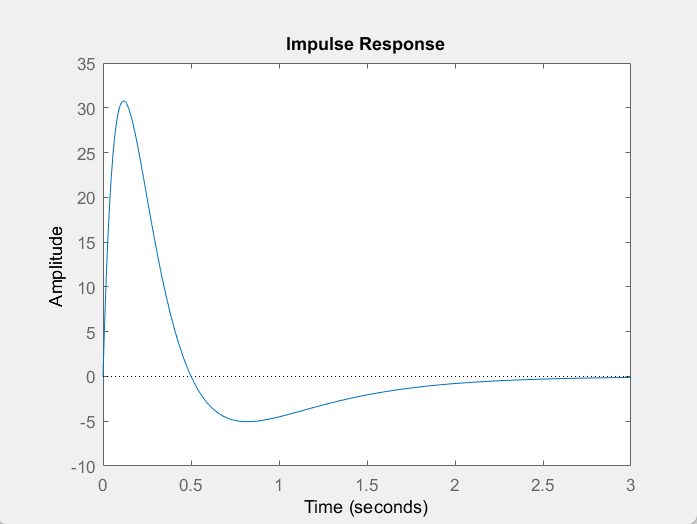


Рисунок 1.1 – График импульсной переходной характеристики

Для получения аналитической формы переходной характеристики дополним систему интегратором:

|  |  |
| --- | --- |
|  | (1.9) |

С помощью метода неопределенных коэффициентов найдем коэффициенты :

Тогда выражение примет вид:

|  |  |
| --- | --- |
|  | (1.10) |

Определим как обратное преобразование Лапласа от :

. (1.11)

(1.12)

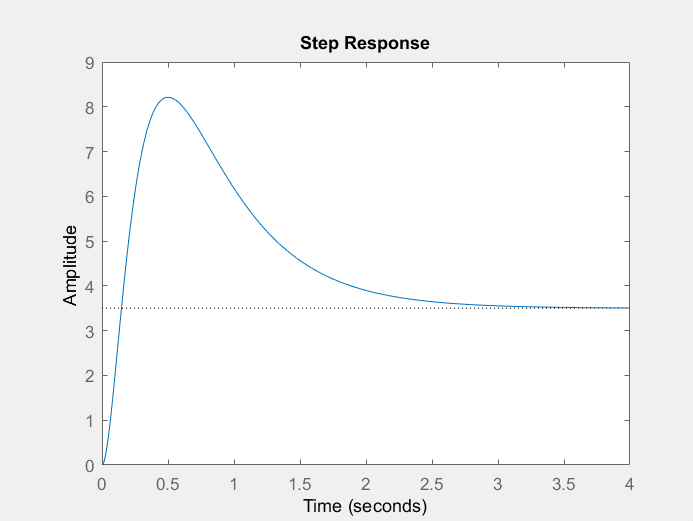


Рисунок 1.2 – График переходной характеристики

Вид переходной характеристики построенный в пакете Matlab представлен на рисунке 1.2.

Система при воздействии на нее импульсного сигнала со временем возвращается в исходное состояние. При воздействии ступенчатого сигнала со временем система приходит в однозначное состояние. Следовательно, заданная по условию система является устойчивой [1].

## **1.2 Построение в МАТLAB частотных характеристик и вид асимптотических ЛАЧХ и ЛФЧХ**

Амплитудно-частотная характеристика (АЧХ) показывает, как изменяется отношение выходного сигнала к входному в зависимости от частоты. Фазочастотная характеристика (ФЧХ) показывает изменение сдвига фаз между входным и выходным сигналами в зависимости от частоты [1].

Преобразуем передаточную функцию к следующему виду:

|  |  |
| --- | --- |
|  | (1.13) |

Передаточная функция представляет собой произведение трех апериодических звеньев и одного форсирующего звена.

(1.14)

Найдем сопрягающие частоты звеньев и коэффициент усиления:

|  |  |
| --- | --- |
|  | (1.15) |

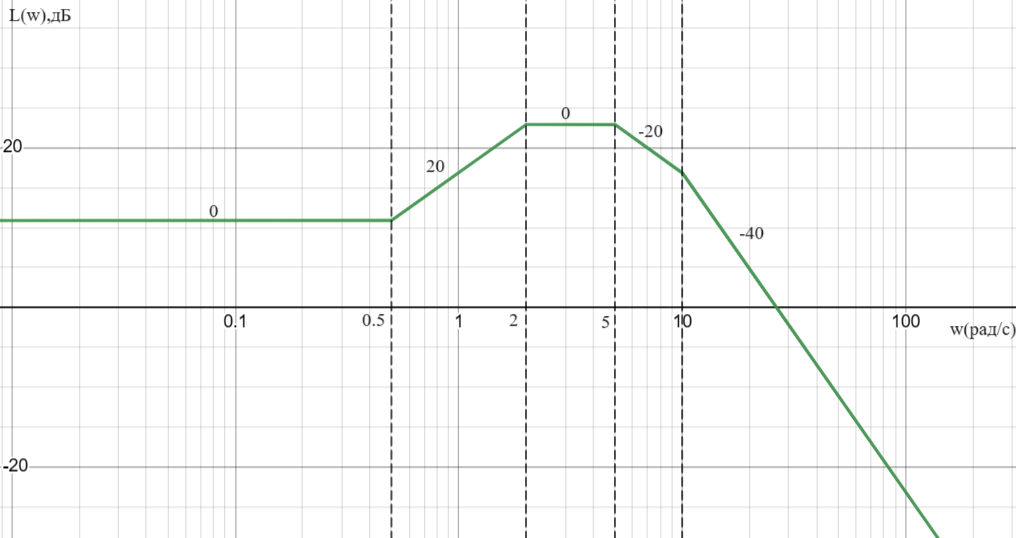
. (1.16)

Фазочастотная характеристика примет вид:

|  |  |
| --- | --- |
|  | (1.17) |

Используя найденные значения коэффициента усиления и сопрягающих частот, построим графики ЛАЧХ и ЛФЧХ. Графики ЛАЧХ и ЛФЧХ представлен на рисунке 1.3 и рисунки 1.4. Графики ЛАЧХ и ЛФЧХ, построенные в пакете Matlab представлены на рисунке 1.5.

Построенные вручную характеристики подобны построенным в пакете Matlab.

Рисунок 1.3 – Логарифмическая амплитудно-частотная характеристика

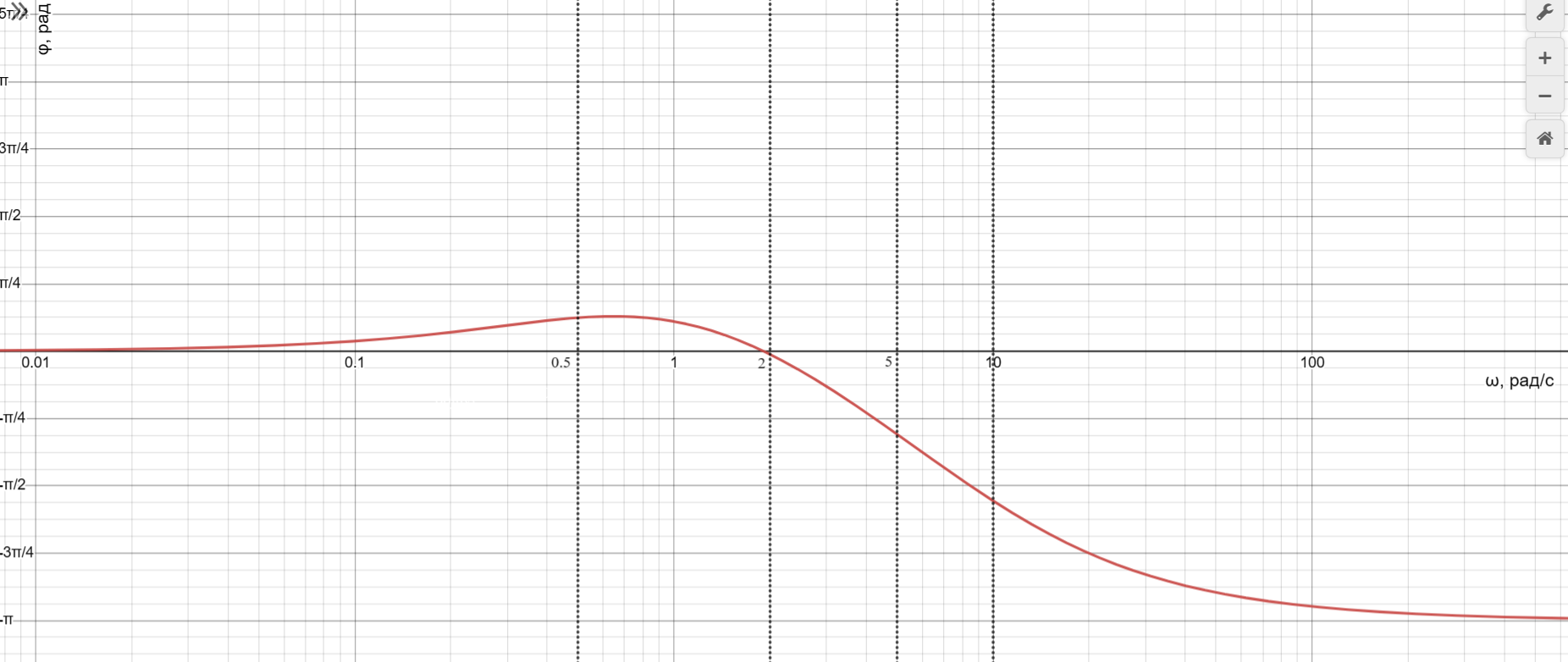


Рисунок 1.4 – Логарифмическая фазо-частотная характеристика

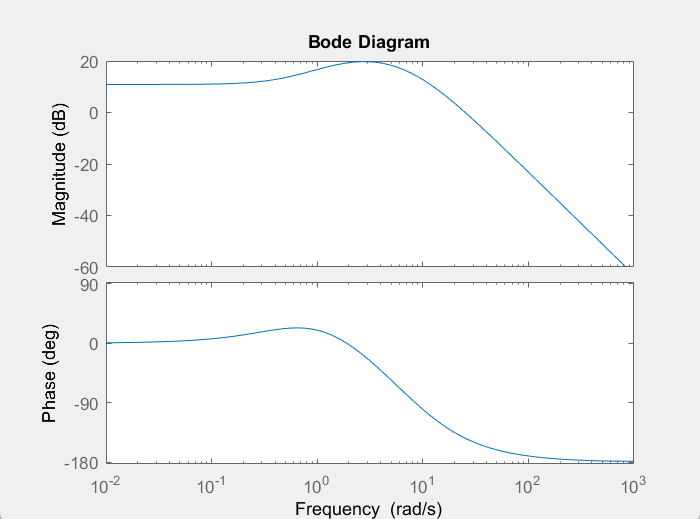


Рисунок 1.5 – Графики частотных характеристик в Matlab

## **1.3 Составление уравнений состояния в нормальной и канонической формах. Результаты моделирования системы**

Кроме входных и выходных переменных при описании систем выделяют переменные , связанные с внутренней структурой устройства – переменные состояния. Тогда систему можно описать с помощью уравнений состояния [3].

Нормальная форма уравнений состояния имеет вид:

(1.18)

Здесь – квадратная матрица, размер которой определяется порядком дифференциального уравнения. Элементы, стоящие над главной диагональю – единицы, элементы нижней строки – коэффициенты левой части дифференциального уравнения, взятые с противоположным знаком. Все остальные элементы – нули. Такая матрица называется матрицей Фробениуса.

Согласно выражению (1.1) дифференциальное уравнение системы имеет вид:

(1.19)

где и – коэффициенты уравнения.

Элементы матриц B и D вычисляются по следующим рекуррентным соотношениям:

.

Поставив рассчитанные матрицы в систему (1.18), получим:

Схема модели в пространстве состояний в нормальной форме, построенная вручную, представлена на рисунке 1.6.

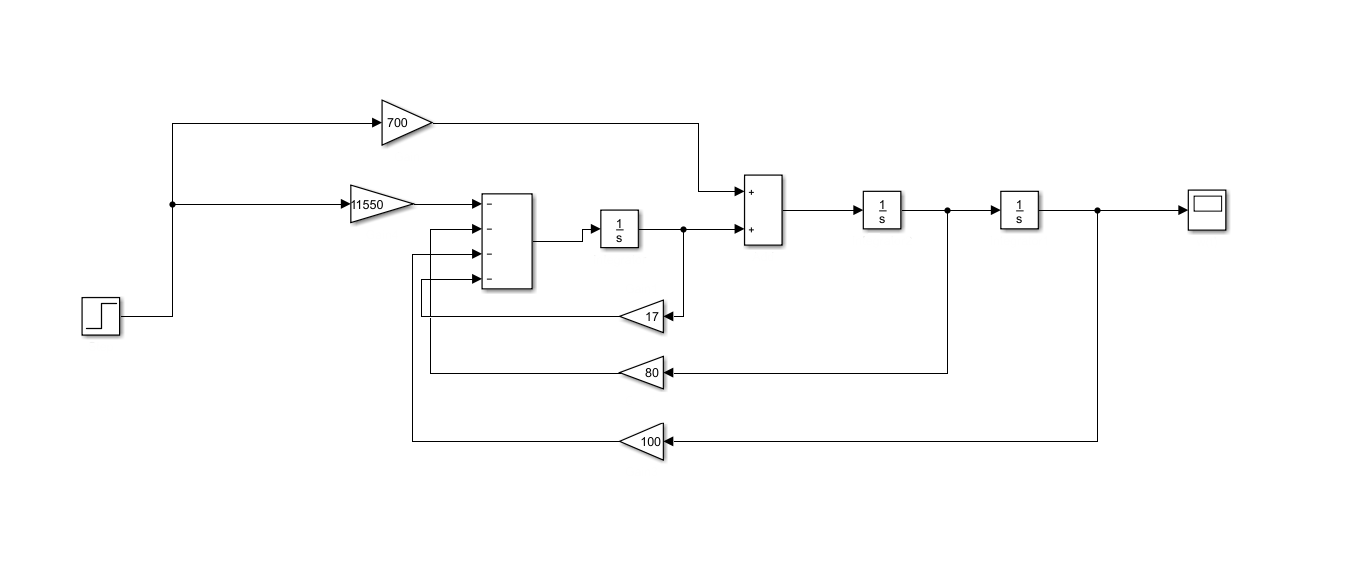


Рисунок 1.6 – Схема модели в нормальной форме

Построим и промоделируем схему модели в нормальной форме в Simulink. Результат моделирования представлен на рисунке 1.7.

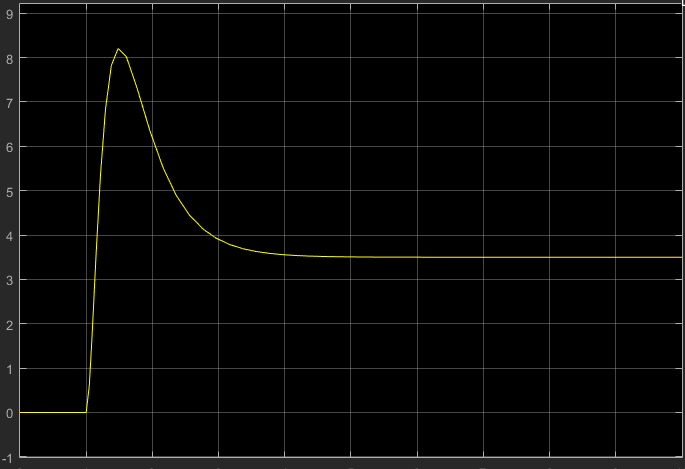


Рисунок 1.7 – Результат моделирования модели в нормальной форме в Simulink

Запишем уравнения состояния в канонической форме. Для этого ведем новую переменную состояния , которая связана с переменной состояния следующим образом: . – это модальная матрица, которая имеет вид:

где –характеристические числа матрицы Фробениуса .

Рассчитаем с помощью матрицы , элементами которой являются характеристические числа матрицы A

Получим уравнение = 0

-2;-5;-10.

Уравнения состояния системы в канонической форме имеют вид:

где – диагональная матрица вида:

,

где – матрица обратная матрице .

Поставив рассчитанные матрицы в систему (1.20), получим:

(1.21)

Схема модели в пространстве состояний в канонической форме построенные вручную представлена на рисунке 1.8.

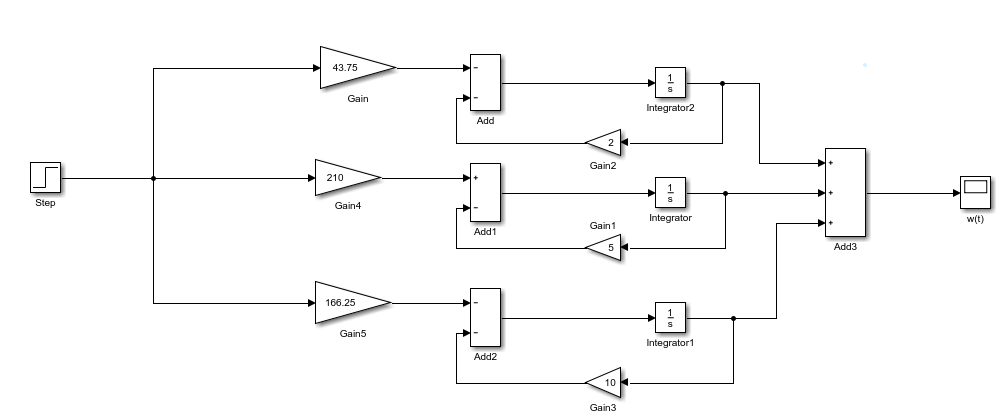


Рисунок 1.8 – Схема модели в канонической форме

Построим и промоделируем модель в канонической форме в Simulink. Результат моделирования представлен на рисунке 1.9.

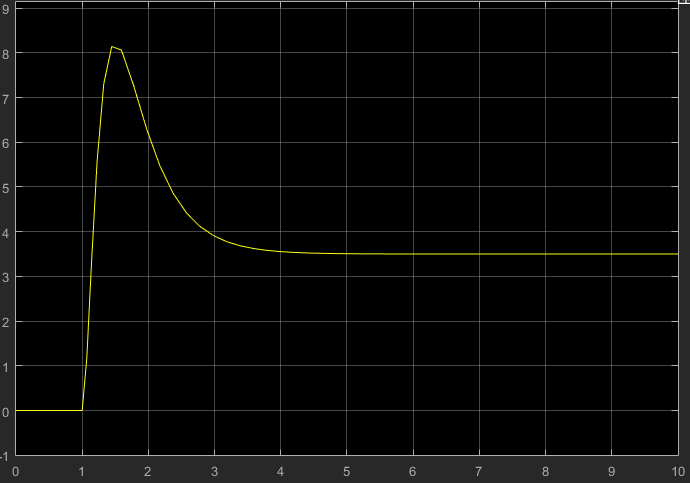


Рисунок 1.9 – Результат моделирования модели в канонической форме в Simulink

Вид переходного процесса для нормальной и канонической форм совпадает.

## 

## 1.4 Решение уравнений состояния в канонической форме

Решим уравнение состояния (1.21), представленное в канонической форме. Каждое из дифференциальных уравнений первого порядка зависит только от одной переменной и его решение в общем виде имеет вид:

|  |  |
| --- | --- |
|  | (1.22) |

Определим начальные условия для вектора :

Найдем выражения для :

Выполним проверку:

;

Проверим, одинаково ли значение коэффициента усиления:

Для передаточной функции (1.1):

Для переходной функции (1.12):

По модели в пространстве состояний в канонической форме:

По аналитической записи импульсной переходной характеристики (1.8):

.

|  |  |
| --- | --- |
|  |  |

Значения коэффициента усиления совпадают.

# 2 Линейное программирование

## **2.1 Расчет оптимального плана и экстремального значения функции цели**

К задачам линейного программирования относятся задачи нахождения условного экстремума функции нескольких переменных, при условии, что функция и ограничения линейны [2].

Общий вид задачи линейного программирования на поиск минимума:

где – матрица из коэффициентов при переменных ограничений;

– вектор-столбец свободных членов в ограничениях;

– вектор-строка коэффициентов при переменных функции цели.

Условие задачи:

(2.1)

Решим задачу (2.1) с помощью симплекс-метода.

Все ограничения системы (2.1) со знаком «» умножим на :

(2.2)

Введем в систему (2.2) дополнительные переменные для ограничений вида неравенств, чтобы преобразовать их в равенства. Для ограничения вида равенства воспользуемся методом искусственного базиса и введем искусственную переменную :

(2.3)

В связи с вводом искусственных переменных функция цели примет вид:

, (2.4)

где – коэффициент штрафа за введение искусственных переменных.

Выразим из ограничения системы:

+ 21

и подставим в выражение (2.4):

(2.5)

При составлении первой симплекс-таблицы будем полагать, что исходные переменные , являются небазисными, а введенные переменные – базисными. В задачах минимизации знак коэффициентов при небазисных переменных в - и -строках не изменяется на противоположный. Знак постоянной величины в -строке изменяется. Оптимизация проводится сначала по -строке. Выбор ведущих столбца и строки, все симплексные преобразования осуществляются как в обычном симплекс-методе.

Шаг 1. Составим начальную симплекс таблицу:

Таблица 2.1 – Первая итерация

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| БП | Своб. члены | НП | | | |  |
|  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |

Решение не является допустимым, так как существуют свободные члены, которые меньше нуля.

Шаг 2.Выберем строку , и выберем в ней максимальный по абсолютному значению отрицательный элемент, который станет ведущим. Строка будет исключена из базиса, а столбец будет включен в базис.

Максимальный по абсолютному значению элемент строки соответствует столбцу . Столбец будет исключен из базиса. Ведущий элемент выделен полужирным шрифтом в таблице 2.1.

Пересчитаем таблицу в соответствии с правилами.

Искусственные переменные, исключенные из базиса, в него больше не возвращаются, поэтому столбцы элементов таких переменных опускаются.

Таблица 2.2 – Вторая итерация

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| БП | Своб. члены | НП | | | |  |
|  |  |  |  |  |
|  | 147/4 | - 3/4 | 9/4 | 6/1 | 4/1 |  |
|  | 0/1 | 1/1 | - 1/1 | - 1/1 | 2/1 |  |
|  | 21/4 | - 1/4 | - 1/4 | 1/1 | 0/1 |  |
|  | 99/4 | - 3/4 | **5/4** | 7/1 | 0/1 |  |
|  | - 21/1 | 1/1 | - 3/1 | 0/1 | - 1/1 |  |
|  | -147/4 | 3/4 | - 9/4 | - 6/1 | - 4/1 |  |

Таблица 2.3 – Третья итерация

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| БП | Своб. члены | НП | | | |  |
|  |  |  |  |  |
|  | - 39/5 | 3/5 | **- 9/5** | - 33/5 | 4/1 |  |
|  | 99/5 | 2/5 | 4/5 | 23/5 | 2/1 |  |
|  | 51/5 | - 2/5 | 1/5 | 12/5 | 0/1 |  |
|  | 99/5 | - 3/5 | 4/5 | 28/5 | 0/1 |  |
|  | 192/5 | - 4/5 | 12/5 | 84/5 | - 1/1 |  |
|  | 39/5 | - 3/5 | 9/5 | 33/5 | - 4/1 |  |

Таблица 2.4 – Четвёртая итерация

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| БП | Своб. члены | НП | | |  | |
|  |  |  |  |
|  | 13/3 | - 1/3 | 11/3 | - 20/9 |  |
|  | 49/3 | 2/3 | 5/3 | 34/9 |  |
|  | 28/3 | - 1/3 | 5/3 | 4/9 |  |
|  | 49/3 | - 1/3 | 8/3 | 16/9 |  |
|  | 28/1 | 0/1 | 8/1 | 13/3 |  |

В исходную функцию цели и ограничения входят только переменные , поэтому оптимальный план решения задачи:

|  |  |
| --- | --- |
|  | (2.6) |

Экстремальное значение функции (2.1) примет значение:

## **2.2 Исследование двойственной задачи линейного программи­рования**

Предположим, что у нас есть прямая задача вида:

Приведем условие к стандартной форме. Так как требуется найти минимум целевой функции, то неравенства в системе ограничений должны быть вида ≥. Третье и четвертое неравенства умножим на (-1), тогда получим:

(2.7)

Составим двойственную задачу для задачи (2.1):

(2.8)

Преобразуем ограничения неравенств в равенства:

(2.9)

Составим симплекс таблицу, используя выражения (2.8) и (2.9):

Таблица 2.5 – Первая итерация

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| БП | Своб. члены | НП | | | |
|  |  |  |  |
|  | 4 | -3 | 4 | 4 | 3 |
|  | -4 | 3 | 2 | -1 | -2 |
|  | 4 | **3** | -3 | 4 | -4 |
|  | -1 | 4 | -2 | 0 | 0 |
|  | 0 | -21 | 21 | -21 | 9 |

Таблица 2.6 – Вторая итерация

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| БП | Своб. члены | НП | | | |
|  |  |  |  |
|  | 8/1 | 1/1 | 1/1 | **8/1** | - 1/1 |
|  | - 8/1 | - 1/1 | 5/1 | - 5/1 | 2/1 |
|  | 4/3 | 1/3 | - 1/1 | 4/3 | - 4/3 |
|  | - 19/3 | - 4/3 | 2/1 | - 16/3 | 16/3 |
|  | 28/1 | 7/1 | 0/1 | 7/1 | - 19/1 |

Таблица 2.7 – Третья итерация

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| БП | Своб. члены | НП | | | |
|  |  |  |  |
|  | 1/1 | 1/8 | 1/8 | 1/8 | - 1/8 |
|  | - 3/1 | **- 3/8** | 45/8 | 5/8 | 11/8 |
|  | 0/1 | 1/6 | - 7/6 | - 1/6 | - 7/6 |
|  | - 1/1 | - 2/3 | 8/3 | 2/3 | 14/3 |
|  | 21/1 | 49/8 | - 7/8 | - 7/8 | -145/8 |

Таблица 2.8 – Четвертая итерация

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| БП | Своб. члены | НП | | | |
|  |  |  |  |
|  | 0/1 | 1/3 | 2/1 | 1/3 | 1/3 |
|  | 8/1 | - 8/3 | - 15/1 | - 5/3 | - 11/3 |
|  | - 4/3 | 4/9 | 4/3 | 1/9 | - 5/9 |
|  | 13/3 | - 16/9 | - 22/3 | - 4/9 | 20/9 |
|  | - 28/1 | 49/3 | 91/1 | 28/3 | 13/3 |

Решение является допустимым (допуская ), и является оптимальным.

Из симплекс таблицы 2.6 получим:

В исходную функцию цели и ограничения входят только переменные , поэтому оптимальный план решения задачи:

|  |  |
| --- | --- |
|  |  |

Экстремальное значение функции (2.8) примет значение:

Переменным прямой задачи поставим в соответствие переменные двойственной задачи:

В -строке симплекс таблицы 2.6 двойственной задачи расположены коэффициенты при небазисных переменных . Используя соответствие, найдем оптимальное решение прямой задачи:

|  |  |
| --- | --- |
|  |  |

Тогда оптимальный план прямой задачи:

|  |  |
| --- | --- |
| Оптимальный план прямой задачи, найденный путем решения двойственной задачи, совпадает с оптимальным планом в выражении (2.6), полученным при решении прямой задачи. Экстремальные значения функции цели прямой и двойственной задачи совпадают. |  |

## **2.3 Решение целочисленной задачи по методу Гомори**

Если решение задачи не является целочисленным, получить частично целочисленное решение путем введения дополнительных ограничений по методу Гомори. Для решения задачи воспользуемся симплекс-таблицей 2.4, решение которой является оптимальным, но не целочисленным:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| БП | Своб. члены | НП | | |  | |
|  |  |  |  |
|  | 13/3 | - 1/3 | 11/3 | - 20/9 |  |
|  | 49/3 | 2/3 | 5/3 | 34/9 |  |
|  | 28/3 | - 1/3 | 5/3 | 4/9 |  |
|  | 49/3 | - 1/3 | 8/3 | 16/9 |  |
|  | 28/1 | 0/1 | 8/1 | 13/3 |  |

Из выражения (2.6):

{-} + {}+ {} {}

+ +

Домножим на (-1) и приведём к равенству, вводя дополнительную переменную :

Таблица 2.9 – Первая итерация

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| БП | Своб. члены | НП | | |  | |
|  |  |  |  |
|  | 13/3 | - 1/3 | 11/3 | - 20/9 |  |
|  | 49/3 | 2/3 | 5/3 | 34/9 |  |
|  | 28/3 | - 1/3 | 5/3 | 4/9 |  |
|  | 49/3 | - 1/3 | 8/3 | 16/9 |  |
|  | -1/3 | -2/3 | -2/3 | **-4/9** |  |
|  | 28/1 | 0/1 | 8/1 | 13/3 |  |

Таблица 2.10 – Вторая итерация

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| БП | Своб. члены | НП | | |  | |
|  |  |  |  |
|  | 6/1 | 3/1 | 7/1 | - 5/1 |  |
|  | 27/2 | - 5/1 | - 4/1 | 17/2 |  |
|  | 9/1 | - 1/1 | 1/1 | 1 |  |
|  | 15/1 | - 3/1 | 0/1 | 4/1 |  |
|  | 3/4 | 3/2 | 3/2 | -9/4 |  |
|  | -125/4 | - 13/2 | 3/2 | 39/4 |  |

Из таблицы 2.10 получили: 𝐹 = , = 9, = 15. Мы получили полностью целочисленное решение для переменной 𝑥.

# 3 Нелинейное программирование

## **3.1 Нахождение безусловного экстремума функции *F*(*x*)**

Исходная задача имеет вид:

(3.1)

Начальная точка имеет координаты:

График функции, построенный в Matlab, представлен на рисунке 3.1.

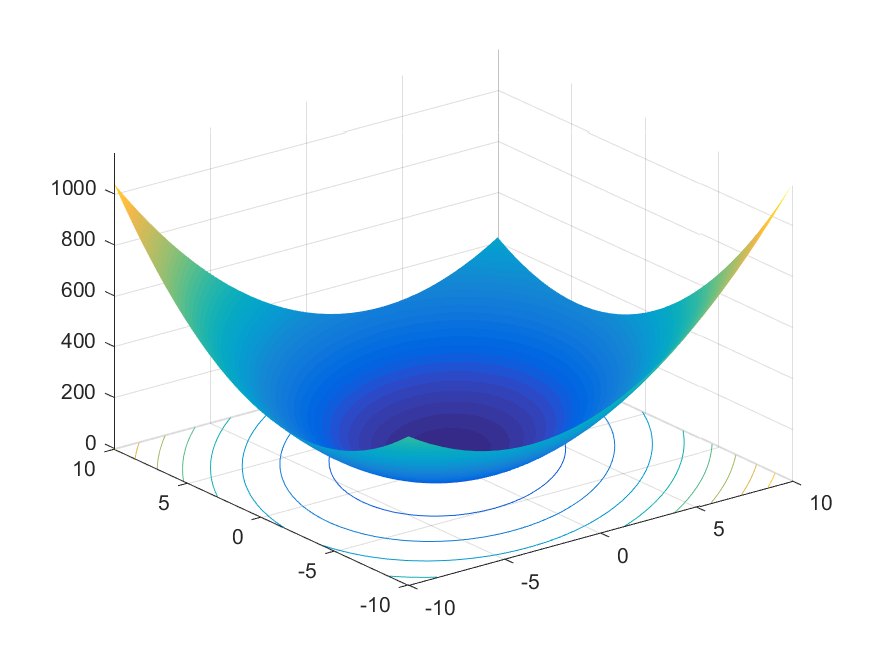


Рисунок 3.1 – График функции в Matlab

Решим задачу различными методами и сравним полученные результаты.

Метод Ньютона-Рафсона.

В данном методе решение заданной нелинейной задачи, как правило, происходит за один шаг, т.е. будет решением данной задачи.

Здесь – матрица Гессе (матрица, составленная из вторых частных производных), – значение градиента функции в начальной точке.

Найдем вид вектора градиента:

|  |  |
| --- | --- |
|  | (3.2) |

В точке вектор градиента примет значение:

Составим матрицу Гессе:

Найдем обратную матрицу для матрицы Гессе.

Координаты следующей точки будут определятся по выражению:

Найдем значение вектора градиента по выражению (3.2) в точке :

Следовательно в точке функция достигает своего максимального значения:

Метод наискорейшего спуска

В данном методе на каждой итерации в текущей точке определяется направление движения (вектором градиента для задачи на максимум) и величина шага в данном направлении [2].

*Шаг 1.*

Координаты точки будут определяться выражением:

где – значение вектора градиента, вычисленное в точке ;

– величина шага в данном направлении.

Найдем значение функции по выражению (3.1) в точке :

Найдем направление вектора градиента по выражению (3.2) в точке :

Подставим известные значения в выражение для определения координаты следующей точки:

Найдем величину шага . Для этого подставим в функцию (3.1) найденные выражения для , т.е. получим функцию зависящую от величины шага. Затем исследуем полученную функцию на экстремум, для чего возьмем производную от полученной функции и приравняем к нулю:

Тогда координаты точки будут равны:

Найдем значение функции по выражению (3.1) в точке :

*Шаг 2.*

Координаты точки будут определяться выражением:

Найдем направление вектора градиента по выражению (3.2) в точке :

Подставим известные значения в выражение для определения координаты следующей точки:

Найдем величину шага . Для этого подставим в функцию (3.1) найденные выражения для , т.е. получим функцию зависящую от величины шага. Затем исследуем полученную функцию на экстремум:

Тогда координаты точки будут равны:

Найдем значение функции по выражению (3.1) в точке :

*Шаг 3.*

Координаты точки будут определяться выражением:

Найдем направление вектора градиента по выражению (3.2) в точке :

Подставим известные значения в выражение для определения координаты следующей точки:

Найдем величину шага :

Тогда координаты точки будут равны:

Найдем значение функции по выражению (3.1) в точке :

Графическая интерпретация метода найскорейшего спуска представлена на рисунке 3.2.

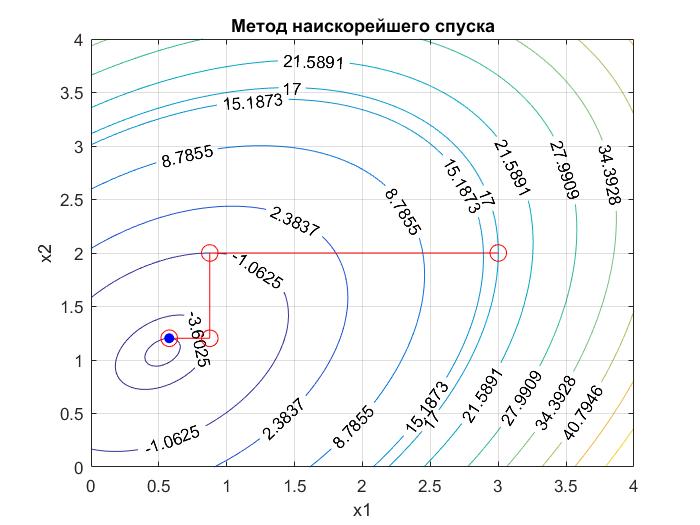


Рисунок 3.2 – Графическая интерпретация метода наискорейшего спуска

Метод наискорейшего спуска для данной функции медленно сходится к точному решению, что видно из расчетов и рисунка.

## **3.2 Нахождение экстремума функции *F*(*x*) с учетом системы ограничений**

На задачу (3.1) наложим ограничения на значения переменных в соответствии с условием. Полученная задача примет вид:

(3.3)

Область допустимых значений переменных построена на рисунке 3.4.

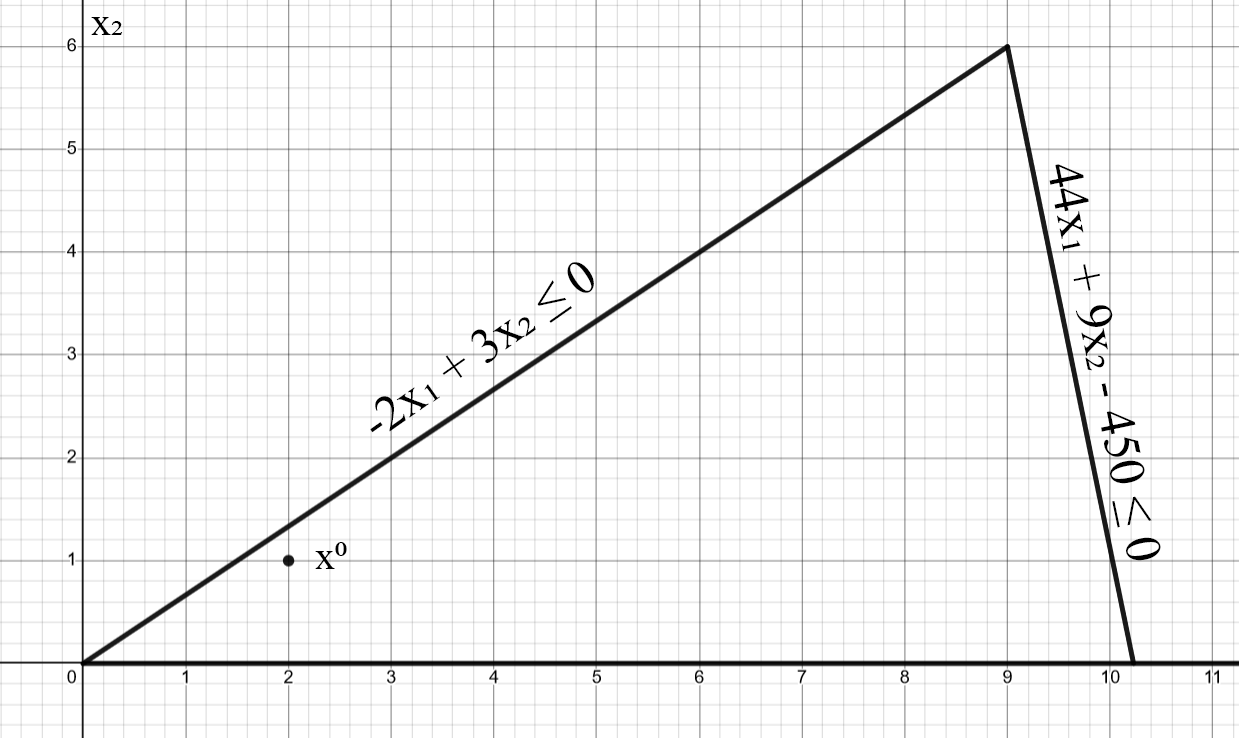


Рисунок 3.4 – Область допустимых значений переменных

Метод допустимых направлений Зойтендейка

Метод Зойтендейка является расширением метода наискорейшего спуска, позволяющий учитывать ограничения. На каждом шаге строится возможное допустимое направление шага, и выбирается величина шага в соответствии с ограничениями.

*Шаг 1*

Координаты точки задаются выражением:

Найдем направление вектора градиента по выражению (3.2) в точке :

Подставляя в выражение для , получим:

Найдем величину шага :

Найдем интервал допустимых значений *.* Для этого подставим в ограничения (3.3) найденные выражения для :

Найденное не входит в найденный выше интервал. В качестве величины шага возьмем правую границу интервала .

Подставляя в выражение для , получим:

*Шаг 2*

Найдем направление вектора градиента по выражению (3.2) в точке :

Вектор антиградиента направлен за ОДЗП. Поэтому необходимо найти направление , в сторону которого нужно двигаться. При этом координаты точки задаются выражением:

Найдем направление . Так как точка принадлежит граничной прямой ограничения 1 системы (3.3), то направление очередного шага определяется из условия:

Откуда следует, что . Тогда из условия нормировки:

Скалярное произведение векторов и :

( при

( при

Следовательно, выбираем направление , так как оно составляет острый угол с вектором антиградиента.

Подставляя в выражение для , получим:

Найдем величину шага :

Найдем интервал допустимых значений , при этом исключим из системы (3.3) ограничение 1, на границе которого находится точка :

Найденное входит в найденный выше интервал.

Подставляя в выражение для , получим:

*Шаг 3*

Найдем направление вектора градиента по выражению (3.2) в точке :

Скалярное произведение направление составляет прямой угол с вектором антиградиента. Следовательно, оптимальное решение найдено.

Значение функции в точке :

Графическая интерпретация задачи представлена на рисунке 3.5.

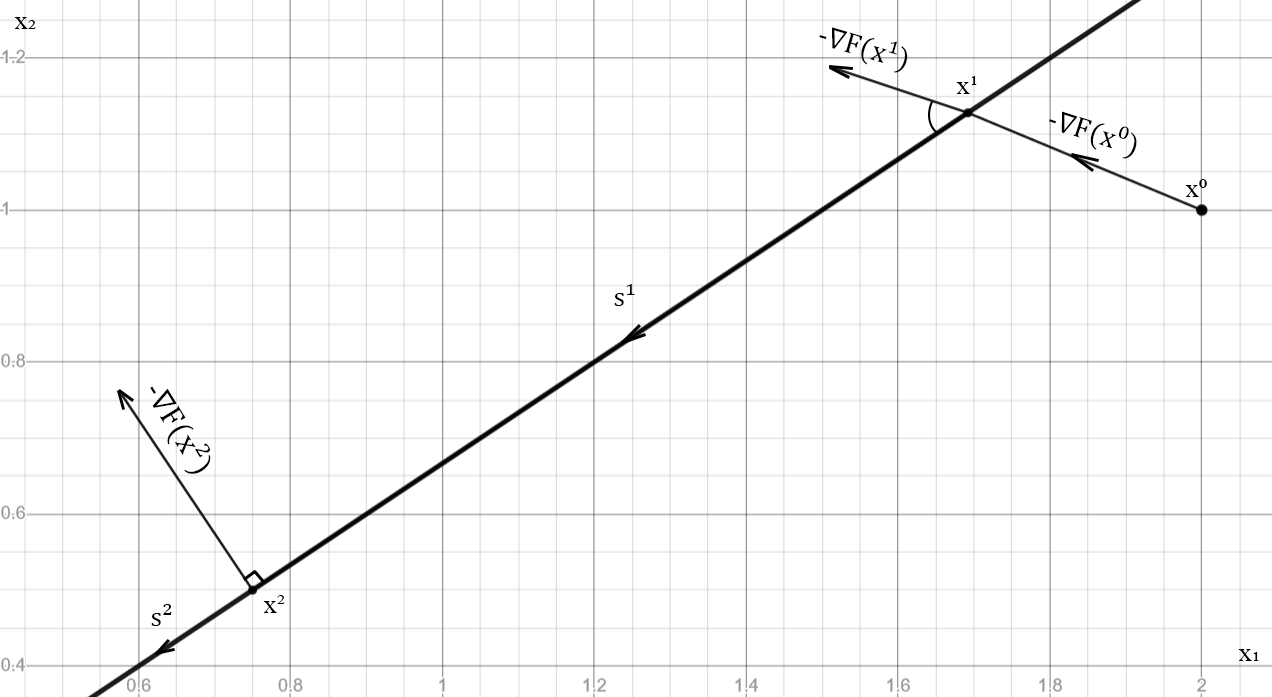


Рисунок 3.5 – Графическая интерпретация метода Зойтендейка

Метод Куна-Таккера.

Метод предназначен для решения задачи, в которой функция является квадратичной, а все ограничения линейны.

Метод основан на использовании теоремы Куна-Таккера.

Функция Лагранжа имеет вид:

(3.3)

где – неопределенные множители Лагранжа;

Условия теоремы Куна-Таккера для задачи на поиск минимума:

|  |  |
| --- | --- |
|  | (3.4) |

Составим функцию Лагранжа для задачи:

|  |  |
| --- | --- |
|  | (3.5) |

Составим систему уравнений в соответствии с выражением (3.4):

|  |  |
| --- | --- |
|  | (3.6) |

Приведем ограничения задачи (3.6) к виду равенств, введя дополнительные переменные :

|  |  |
| --- | --- |
|  | (3.7) |

Для решения задачи линейного программирования (3.7) составим симплекс-таблицу.

Таблица 3.1 – Первая итерация

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| БП | Своб. члены | НП | | | |
|  |  |  |  |
|  | - |  | 3 |  |  |
|  | -7 | 3 |  | -3 |  |
|  | 0 | -2 |  | 0 | 0 |
|  |  | 44 | 9 | 0 | 0 |

Таблица 3.2 – Вторая итерация

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| БП | Своб. члены | НП | | | |
|  |  |  |  |
|  | - 29/8 | **- 55/8** | 3/8 | 7/8 | -379/8 |
|  | 7/8 | - 3/8 | - 1/8 | 3/8 | 9/8 |
|  | - 21/8 | - 7/8 | 3/8 | -1,125 | -3,375 |
|  | 3537/8 | 379/8 | 9/8 | - 27/8 | - 81/8 |

Таблица 3.3 – Третья итерация

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| БП | Своб. члены | НП | | | |
|  |  |  |  |
|  | 29/55 | -8/55 | -3/55 | -7/55 | 379/55 |
|  | 59/55 | -3/55 | -8/55 | 18/55 | 204/55 |
|  | -199/55 | -7/55 | 18/55 | **-1,236** | 2.654 |
|  | 417.14 | 379/55 | 204/55 | 146/55 | -336.58 |

Таблица 3.4 – Итоговая симплекс-таблица

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| БП | Своб. члены | НП | | | |
|  |  |  |  |
|  | 3/4 | -9/68 | -3/34 | -7/68 | 225/34 |
|  | 1/2 | -3/34 | -1/17 | 9/34 | 75/17 |
|  | 7/4 | 7/68 | -9/34 | -0.808 | -2.147 |
|  | 825/2 | 225/34 | 75/17 | 73/34 | 5625/17 |

Решения является допустимым, так как все свободные члены положительны.

Из симплекс таблицы 3.2 получим:

|  |  |
| --- | --- |
|  |  |

Искомая точка экстремума .

Решение задачи методом Куна-Таккера совпадает с решением методом линейных комбинаций

Метод линейных комбинаций



В данном методе на каждом шаге в предыдущей точке нелинейная функция цели линеаризуется посредством разложения в ряд Тейлора в окрестности данной точки, пренебрегая всеми степенями старше первой. Затем решается задача линейного программирования, её решение будет в некоторой вершине ОДЗП. После этого необходимо найти величину шага в направлении вершины и координаты следующей точки.

Линеаризованная функция имеет вид:

Здесь является постоянной величиной, поэтому не оказывает влияние на максимизацию. Тогда можно записать:

*Шаг 1*

Зададим точность

Координаты точки задаются выражением:

Найдем направление вектора градиента по выражению (3.2) в точке :

Составим для текущего шага:

Решим задачу линейного программирования:

Составим симплекс таблицу 3.3:

Таблица 3.3 – Первая итерация

|  |  |  |  |
| --- | --- | --- | --- |
| БП | Своб. члены | НП | |
|  |  |
|  | 0 | -2 | **3** |
|  | 450 | 44 | 9 |
|  | 0 | 12 | -5 |

Решение не является допустимым, так как есть отрицательные элемент в -строке.

Выберем столбец , в котором находится максимальный по абсолютному значению отрицательный элемент -строки. Строка имеет минимальное симплекс-отношение и будет исключена из базиса, а столбец будет включен в базис.Выбранный ведущий элемент выделен жирным шрифтом и подчеркнут в таблице 3.3.

Симплекс таблица после пересчета имеет вид, представленный в таблице 3.4.

Таблица 3.4 – Вторая итерация

|  |  |  |  |
| --- | --- | --- | --- |
| БП | Своб. члены | НП | |
|  |  |
|  | 0 | -2/3 | 1/3 |
|  | 450 | 50 | -3 |
|  | 0 | 8.667 | 1.667 |

Решение является допустимым, так как нет отрицательных свободных членов. Решение является оптимальным, так как нет отрицательных элементов в -строке.

Из симплекс-таблицы 3.4 получим:

|  |  |
| --- | --- |
|  |  |

Найденное решение .

Подставляя в выражение для , получим:

Найдем величину шага :

Найденное входит в интервал .

Подставляя в выражение для , получим:

Значение функции в точке :

Проверим близость к решению:

Следовательно, продолжаем поиск точки.

*Шаг 2*

Координаты точки задаются выражением:

Найдем направление вектора градиента по выражению (3.2) в точке :

Составим для текущего шага:

Решим задачу линейного программирования:

Составим симплекс таблицу 3.5:

Таблица 3.5 – Первая итерация

|  |  |  |  |
| --- | --- | --- | --- |
| БП | Своб. члены | НП | |
|  |  |
|  | 0 | -2 | **3** |
|  | 450 | 44 | 9 |
|  | 0 | 3.179 | -6.357 |

Решение не является допустимым, так как есть отрицательные элемент в -строке.

Выберем столбец , в котором находится максимальный по абсолютному значению отрицательный элемент -строки. Строка имеет минимальное симплекс-отношение и будет исключена из базиса, а столбец будет включен в базис.Выбранный ведущий элемент выделен жирным шрифтом и подчеркнут в таблице 3.5.

Симплекс таблица после пересчета имеет вид, представленный в таблице 3.6.

Таблица 3.6 – Первая итерация

|  |  |  |  |
| --- | --- | --- | --- |
| БП | Своб. члены | НП | |
|  |  |
|  | 0 | -2/3 | 1/3 |
|  | 450 | **50** | -3 |
|  | 0 | -1.06 | 2.119 |

Решение не является допустимым, так как есть отрицательные элемент в -строке.

Выберем столбец , в котором находится максимальный по абсолютному значению отрицательный элемент -строки. Строка имеет минимальное симплекс-отношение и будет исключена из базиса, а столбец будет включен в базис.Выбранный ведущий элемент выделен жирным шрифтом и подчеркнут в таблице 3.6.

Симплекс таблица после пересчета имеет вид, представленный в таблице 3.7.

Таблица 3.7 – Вторая итерация

|  |  |  |  |
| --- | --- | --- | --- |
| БП | Своб. члены | НП | |
|  |  |
|  | 6 | 1/75 | 22/75 |
|  | 9 | 1/50 | -3/50 |
|  | 9.536 | 0.021 | 2.055 |

Решение является допустимым, так как нет отрицательных свободных членов. Решение является оптимальным, так как нет отрицательных элементов в -строке.

Из симплекс-таблицы 3.7 получим:

|  |  |
| --- | --- |
|  |  |

Найденное решение .

Подставляя в выражение для , получим:

Найдем величину шага :

Найденное входит в интервал .

Подставляя в выражение для , получим:

Значение функции в точке :

Проверим близость к решению:

Следовательно, продолжаем поиск точки.

*Шаг 3*

Координаты точки задаются выражением:

Найдем направление вектора градиента по выражению (3.2) в точке :

Составим для текущего шага:

Решим задачу линейного программирования:

Составим симплекс таблицу 3.8:

Таблица 3.8 – Первая итерация

|  |  |  |  |
| --- | --- | --- | --- |
| БП | Своб. члены | НП | |
|  |  |
|  | 0 | -2 | **3** |
|  | 450 | 44 | 9 |
|  | 0 | 4.072 | -5.99 |

Решение не является допустимым, так как есть отрицательные элемент в -строке.

Выберем столбец , в котором находится максимальный по абсолютному значению отрицательный элемент -строки. Строка имеет минимальное симплекс-отношение и будет исключена из базиса, а столбец будет включен в базис.Выбранный ведущий элемент выделен жирным шрифтом и подчеркнут в таблице 3.8.

Симплекс таблица после пересчета имеет вид, представленный в таблице 3.9.

Таблица 3.9 – Вторая итерация

|  |  |  |  |
| --- | --- | --- | --- |
| БП | Своб. члены | НП | |
|  |  |
|  | 0 | -2/3 | 1/3 |
|  | 450 | 50 | -3 |
|  | 0 | 0.077 | 2 |

Решение является допустимым, так как нет отрицательных свободных членов. Решение является оптимальным, так как нет отрицательных элементов в -строке.

Из симплекс-таблицы 3.9 получим:

|  |  |
| --- | --- |
|  |  |

Найденное решение .

Подставляя в выражение для , получим:

Найдем величину шага :

Найденное входит в интервал .

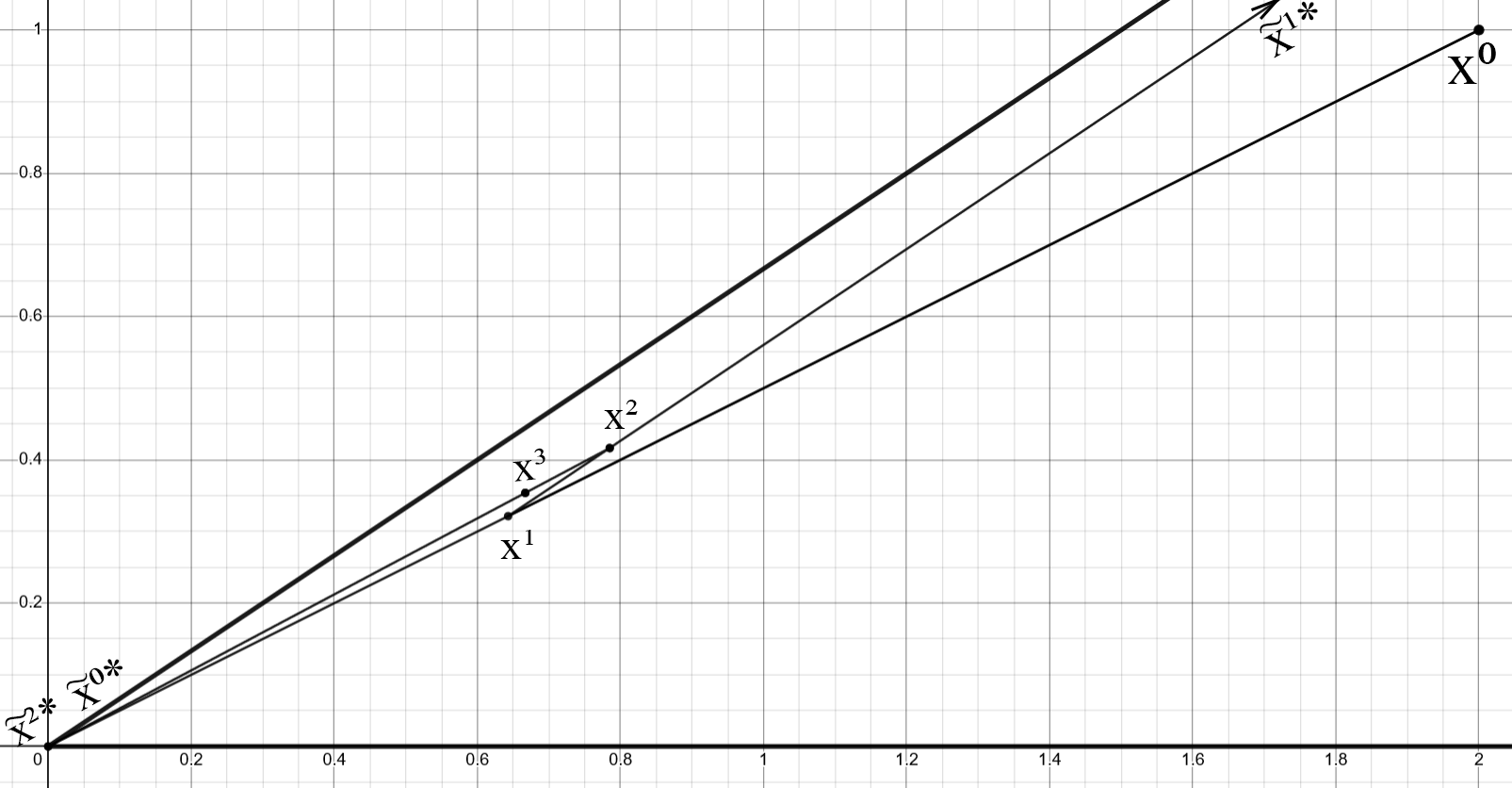
Подставляя в выражение для , получим:

Значение функции в точке :

Проверим близость к решению:

Следовательно, экстремум с заданной точностью найден.

Графическая интерпретация решения задачи методом линейных комбинаций представлена на рисунке 3.6.

Рисунок 3.6 – Графическая интерпретация метода линейных комбинаций

Решение методом линейных комбинаций при большом количестве итераций совпадает с решением методом Зойтендейка и методом Куна-Таккера.

# Заключение

В первой части курсового проекта выполнен анализ линейной системы 3-го порядка, заданной в виде передаточной функции. Получены выражения для построения временных характеристик системы. По заданной передаточной функции были построены логарифмические амплитудно-частотная и фазочастотная характеристики. Правильность результатов построения подтверждена моделированием в пакете Matlab/Simulink.

Также на основании заданной передаточной функции были составлены уравнения состояния в нормальной и канонической формах. Получены схемы моделей системы и проведено моделирование в пакете Matlab/Simulink.

Во второй части курсового проекта решена прямая задача линейного программирования с применением симплекс-таблиц, составлена и решена двойственная задача к прямой. Решение прямой задачи и полученное решение при приведении в соответствие переменных двойственной и прямой задачи совпадает. Также решена частично-целочисленная задача.

В третьей части курсового проекта решены задачи нелинейного программирования без ограничений и с ограничениями. В решении задачи без ограничений показано, что методом Ньютона-Рафсона задача решается за один шаг, а метод наискорейшего спуска медленно сходится к решению. В задаче нелинейного программирования с ограничениями показано, что все методы решения задач одинаково сходятся к одному решению, но за разное количество шагов. Приведены графики интерпретации метода наискорейшего спуска, метода допустимых направлений Зойтендейка и метода линейных комбинаций.

**Литература**

1.  Математические основы теории систем: Конспект лекций для студентов

специальности «Информационные технологии и управление в технических системах». Ч. 2. [Электронный ресурс]. – Минск: БГУИР, 2005. – 145 с. Режим доступа: <http://www.bsuir.by/m/12_100229_1_63815.pdf>.

2.  Математические основы теории систем: Конспект лекций для студентов

специальности «Информационные технологии и управление в технических системах». Ч. 1. [Электронный ресурс]. – Минск: БГУИР, 2005. – 134 с. Режим доступа: https://www.bsuir.by/m/12\_100229\_1\_65744.pdf.