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 27 ПЗ

Студент: гр. 222401 Саркисов Алексей

Руководитель: кандидат технических наук, доцент Павлова А.В.

Минск 2023

Учреждение образования

«Белорусский государственный университет информатики   
и радиоэлектроники»

Факультет информационных технологий и управления

ЗАДАНИЕ

по курсовой работе

Студенту    Саркисову Алексею Владимировичу*–––––––––––––––––––*

1. Тема работы: Математичекие 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)= ; ограничения представлены системой

3.Целевая нелинейная функция F(x)=

; Линейные ограничения имеют вид:

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 5 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*–*

А.В.Павлова*.–––––––А–––ААААа.ВА*

(подпись)

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

(дата и подпись студента)

# Содержание

[Введение 5](#_Toc151931802)

[1 ИССЛЕДОВАНИЕ СИСТЕМ УПРАВЛЕНИЯ 6](#_Toc151931803)

[1.1 Вычисление и построение в Matlab временных характеристик систем 6](#_Toc151931804)

[1.2 Построение асимптотических логарифмических частотных характеристик](#_Toc151931805)……. 11

[1.3 Составление уравнений состояний в нормальной и канонической формах 13](#_Toc151931806)

[1.4 Решение уравнений состояния в канонической форме 20](#_Toc151931807)

[2 Линейное программирование 22](#_Toc151931808)

[2.1 Расчет оптимального плана и экстремального значения функции цели 22](#_Toc151931809)

[2.2 Исследование двойственной задачи линейного программирования 26](#_Toc151931810)

[2.3 Нахождение целочисленного решения задачи 30](#_Toc151931811)

[3 Нелинейное программирование 33](#_Toc151931812)

[3.1 Построение ОДЗП, выбор начальной точки поиска 33](#_Toc151931813)

[3.2 Нахождение экстремального значения функции F(x) без учета ограничений на переменные 35](#_Toc151931814)

[3.2.1 Метод наискорейшего спуска 35](#_Toc151931815)

[3.2.2 Метод Ньютона-Рафсона 37](#_Toc151931816)

[3.2.3 Нахождение в среде MATLAB 38](#_Toc151931817)

[3.3 Нахождение экстремума функции F(x) с учетом системы ограничений 39](#_Toc151931818)

[3.3.1 Метод допустимых направлений Зойтендейка 39](#_Toc151931819)

[3.3.2 Метод линейных комбинаций 41](#_Toc151931820)

[3.3.3 Теорема Куна-Таккера 43](#_Toc151931821)

[3.3.4 Квадратичное программирование в MATLAB 45](#_Toc151931822)

[Заключение 46](#_Toc151931823)

[Список использованных источников…………………………………………………............ 47](#_Toc151931824)

# 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 [2].

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 Вычисление и построение в Matlab временных характеристик систем

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

Передаточная функция имеет вид:

|  |  |
| --- | --- |
|  | (1.1) |
| Для описания моделей систем и действий над ними широко используется система MATLAB и пакет прикладных программ Control System Toolbox. В пакете введен класс объектов, называемый lti объекты – линейные с постоянными параметрами. При создании lti объекта ему присваивается имя. Передаточная функция имеет несколько форм представления:  **tf-форма**, в которой передаточная функция задается двумя векторами строками, составленными из коэффициентов многочленов числителя и знаменателя в порядке убывания степеней s.  tf-форма имеет вид:    **zpk-форма** нулей, полюсов и коэффициента усиления, в которой переда-точная функция (6.5) описывается двумя векторами-строками и одним числом, задающим нули, полюсы и коэффициент усиления системы.  zpk-форма имеет вид:    **ss-форма** представляет передаточную функцию в параметрах пространства состояний.  ss-форма имеет вид: |  |

Характеристическое уравнение системы определяется знаменателем передаточной функции и имеет вид:

. (1.2)

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.

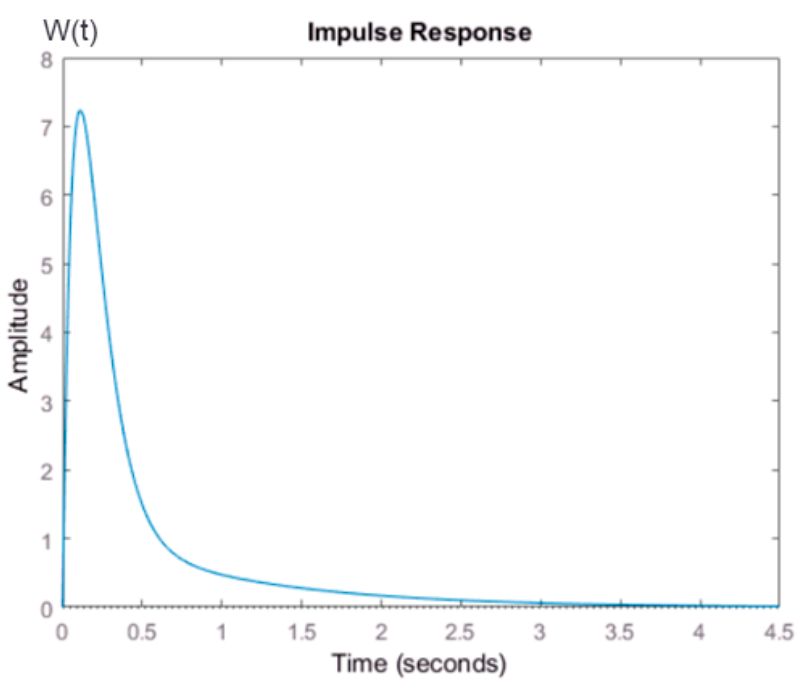


Рисунок 1.1 – Импульсная переходная характеристика

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

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)

Вид переходной характеристики, построенный в пакете MATLAB представлен на рисунке 1.2.

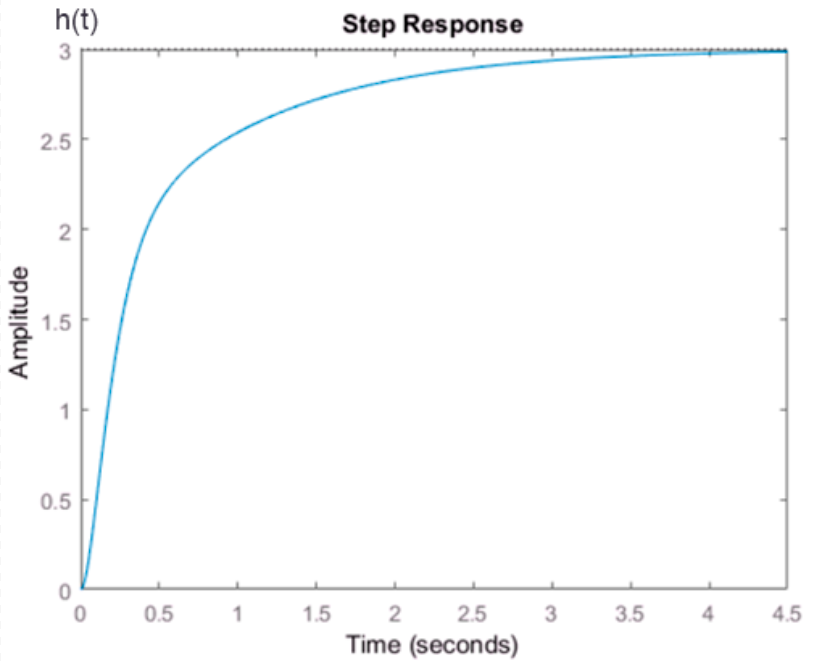


Рисунок 1.2 – Переходная характеристика

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 асимптотических логарифмических частотных характеристик

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 ФЧХ. Графики ЛАЧХ и ФЧХ представлен на рисунке 1.3. Графики ЛАЧХ и ФЧХ, построенные в пакете Matlab представлены на рисунке 1.4.

Построенные вручную характеристики подобны построенным в пакете Matlab.

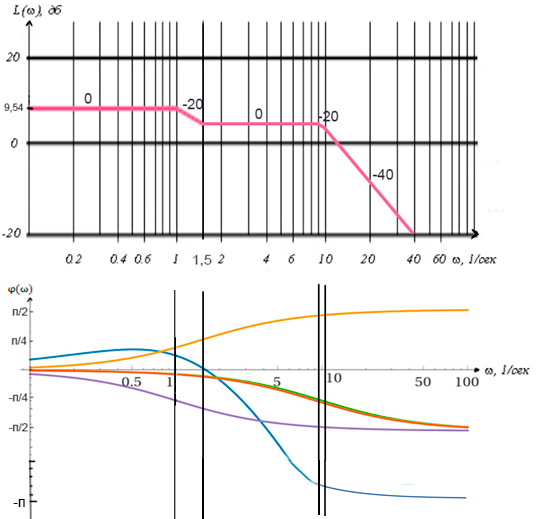


Рисунок 1.4 – Логарифмическая амплитудно-фазовая частотная характеристика(ЛАЧХ) и фазо-частотная характеристика(ФЧХ)

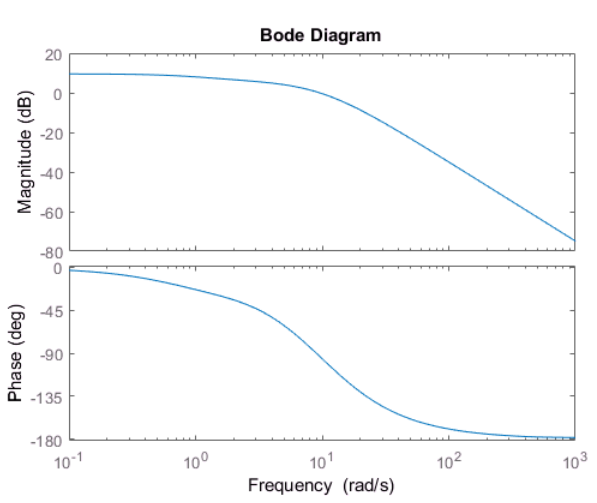


Рисунок 1.4 – Частотные характеристики в MATLAB

## 1.3 Составление уравнений состояний в нормальной и канонической формах

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 1.5.

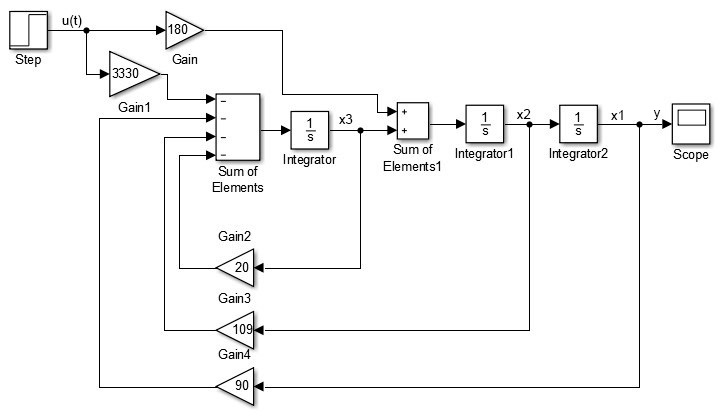


Рисунок 1.5 – Схема модели в пространстве состояний в нормальной форме, построенная вручную

Построим и промоделируем схему модели в нормальной форме в Simulink. Результат моделирования представлен на рисунке 1.6.

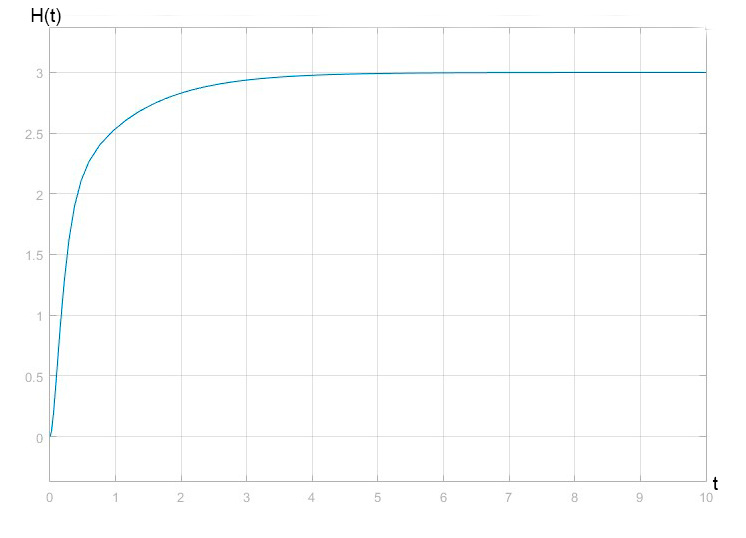


Рисунок 1.6 – Результат моделирования модели в нормальной форме в Simulink

Теперь проверим с ss-формой из MATLAB:

Схема модели в пространстве состояний в нормальной форме, построенная в MATLAB, представлена на рисунке 1.5.1.

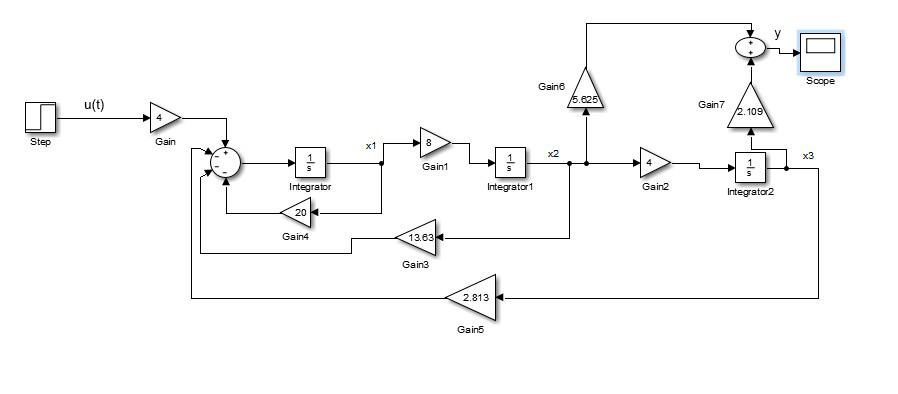


Рисунок 1.5.1 – Схема модели в пространстве состояний в нормальной форме, построенная в MATLAB

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

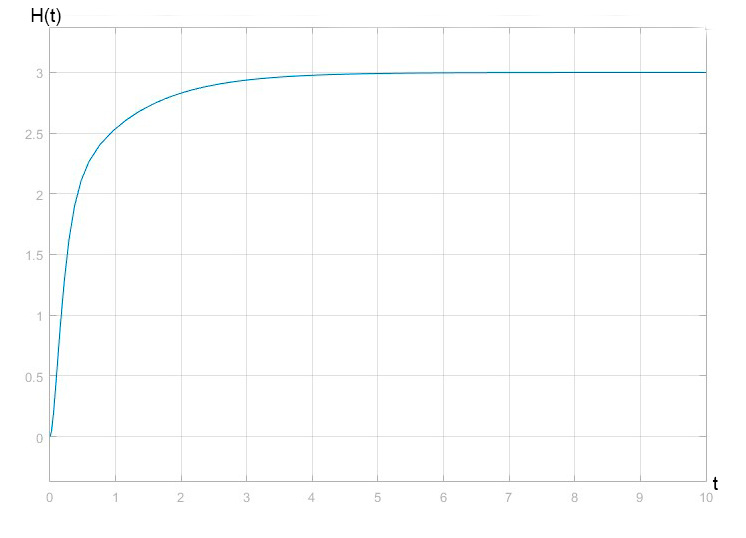


Рисунок 1.7 – Результат моделирования модели ss-формы в Simulink

Как мы видим графики совпадают.

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 .

Уравнения состояния системы в канонической форме имеют вид:

(1.20)

где – диагональная матрица вида:

,

где – матрица обратная матрице .

Поставив рассчитанные матрицы в систему (1.20), получим:

(1.21)

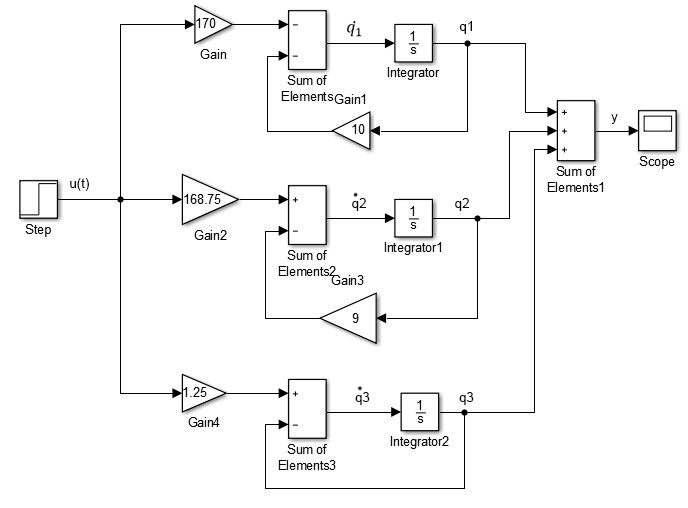
Схема модели в пространстве состояний в канонической форме построенные вручную представлена на рисунке 1.8.

Рисунок 1.8 – Схема модели в пространстве состояний в канонической форме построенная вручную

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

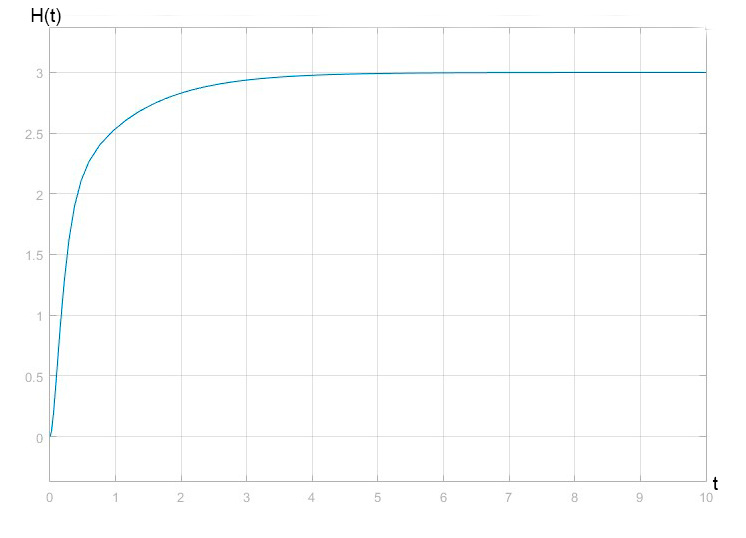


Рисунок 1.9 – Результат моделирования модель в канонической форме в Simulink

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:

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

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 максимума:

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.

Поскольку предстоит решить задачу на нахождение максимума функции цели, то все исходные ограничения должны иметь знак меньше или равно. Для этого 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:

,

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 максимизации знак коэффициентов при небазисных переменных в - и -строках изменяется на противоположный. Знак постоянной величины в -строке не изменяется. Оптимизация проводится сначала по -строке. Выбор ведущих столбца и строки, все симплексные преобразования осуществляются как в обычном симплекс-методе [2].

Шаг 1. Составим начальную симплекс таблицу:

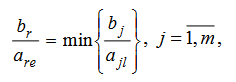
Таблица 2.1 – Первая итерация

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

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

Выберем строку , в которой свободный член меньше нуля, и выберем в ней максимальный по абсолютному значению отрицательный элемент и соответствующий ему столбец определит переменную, которую следует исключить из небазисных и ввести в базис на следующем шаге.

Одновременно из БП исключается та переменная, которая первой изменит знак при увеличении выбранной НП . Это будет , индекс r которой определяется из условия:



Строка, соответствующая переменной , называется ведущей, или разрешающей. Элемент таблицы , стоящий на пересечении ведущей строки и ведущего столбца, называется ведущим, или разрешающим элементом. Нахождением ведущего элемента заканчивается работа с каждой очередной таблицей.

Ведущая переменная выделена полужирным шрифтом.

Пересчитаем таблицу в соответствии с правилами.

Таблица 2.2 – Вторая итерация

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

Таблица 2.3 – Третья итерация

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

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

Из симплекс таблицы 2.3 получим:

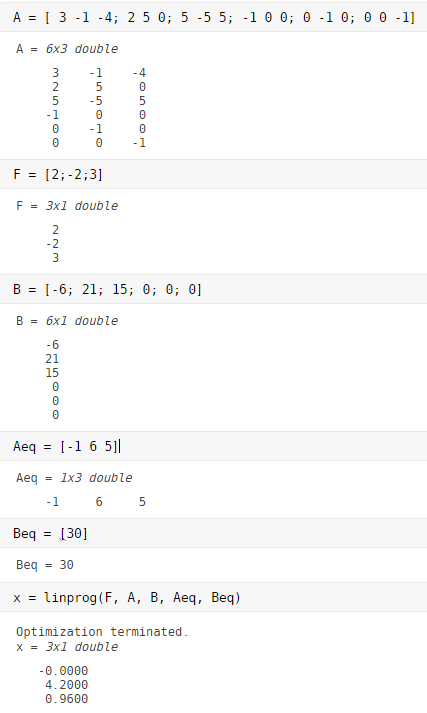
|  |  |
| --- | --- |
|  |  |

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

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

Экстремальное значение функции (2.1) примет значение:

Проверим со значениями в MATLAB:



## 2.2 Исследование двойственной задачи линейного программи­рования

Предположим, что у нас есть прямая задача вида:

Тогда двойственной задачей к этой прямой задаче будет задача вида:

(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 2.4 – Первая итерация

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| БП | Своб. члены | НП | | | |
|  |  |  |  |
|  | 2 | 1 | -3 | -2 | -5 |
|  | -2 | **-6** | 1 | -5 | 5 |
|  | 3 | -5 | 4 | 0 | -5 |
|  | 0 | 30 | -6 | 21 | 15 |

Таблица 2.5 – Вторая итерация

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| БП | Своб. члены | НП | | | |
|  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  | 5 |  |  |  |

Таблица 2.6 – Третья итерация

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| БП | Своб. члены | НП | | | |
|  |  |  |  |
|  |  |  |  |  | -7 |
|  |  |  |  |  |  |
|  | 3 | 0 | 4 |  | -5 |
|  |  |  |  |  |  |

Таблица 2.7 – Четвёртая итерация

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| БП | Своб. члены | НП | | | |
|  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  | 0 |  |  |  |
|  |  |  |  |  |  |

Решение является допустимым и является оптимальным.

Из симплекс таблицы 2.7 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 2.7 двойственной задачи расположены коэффициенты при небазисных переменных . Используя соответствие, найдем оптимальное решение прямой задачи:

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

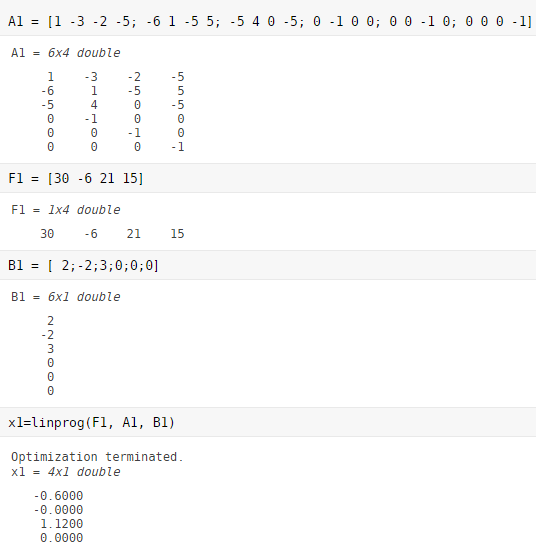
Тогда оптимальный план прямой задачи:

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

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.

Таким образом, переход к двойственной задаче в некоторых случаях может упростить решение за счет уменьшения количества ограничений, а также возможно уменьшение числа шагов при решении двойственной задачи симплекс-методом.

Сравним со значениями из MATLAB:



## 2.3 Нахождение целочисленного решения задачи

Задача, в которой некоторые переменные могут принимать только целые значения, называется частично-целочисленной.

Для задачи (2.1) найдем частично-целочисленное решение, считая, что переменные должны быть целыми.

Дополнительное ограничение должно быть составлено по строке симплекс-таблицы с переменной, значение которой должны быть целочисленными [1]. Дополнительное ограничение имеет вид:

где – коэффициенты при небазисных переменных в данной строке;

– дробная часть свободного члена.

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Добавим условие в симплекс-таблицу: | | | | | | | | | | |
| БП | | Своб. члены | | НП | | | | |
|  | |  | | |
|  | |  | |  | |  | | |
|  | |  | |  | |  | | |
|  | |  | |  | |  | | |
|  | |  | |  | |  | | |
|  | |  | |  | |  | | |
|  | |  | |  | |  | | |
|  | | | | | | | | | | |
| БП | | Своб. члены | | НП | | |
|  | |  |
|  | |  | |  | |  |
|  | |  | |  | |  |
|  | |  | |  | |  |
|  | |  | |  | |  |
|  | |  | |  | |  |
|  | |  | |  | |  |

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

Решение является оптимальным и допустимым.

Из симплекс-таблицы 2.8 получаем:

|  |  |
| --- | --- |
|  | (2.13) |

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

|  |  |
| --- | --- |
|  | (2.14) |

Экстремальное значение функции (2.1) примет значение:

Таким образом, найденное оптимальное решение соответствует требованию целочисленного значения переменной .

# 3 Нелинейное программирование

## 3.1 Построение ОДЗП, выбор начальной точки поиска

Целевая функция имеет вид:





Построим график целевой функции:

>> [x1,x2]=meshgrid([-6:0.1:6]);

>> F=4\*x1.^2+x2.^2+9\*x1+x2

Построим линии уровня функции цели:

>> [c,h] = contour(x1,x2,F)

>> clabel(c,h)

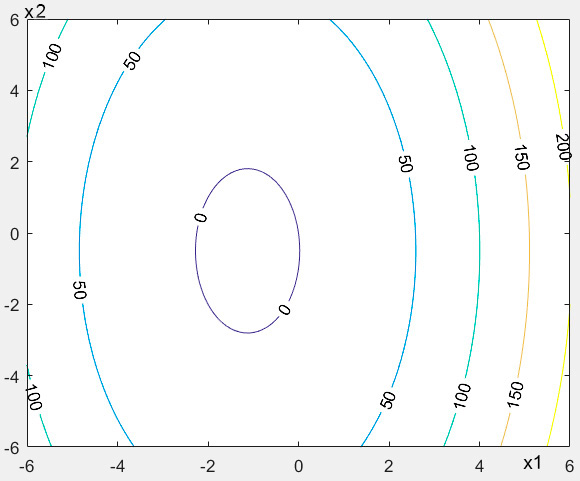


Рисунок 3.1 – Линии уровня функции цели в MATLAB

Построим график функции:

>> meshc(x1,x2,F)

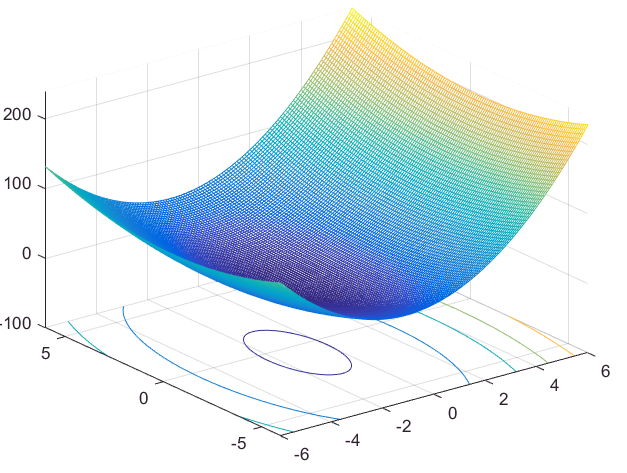


Рисунок 3.2 – График функции в MATLAB

Построим ОДЗП:

>> [x1,x2]=meshgrid([-6:0.1:6]);

>> F=4\*x1.^2+x2.^2+9\*x1+x2

>> hold on

>> ax = gca;

>> ax.XAxisLocation = 'origin';

>> ax.YAxisLocation = 'origin';

>> grid on  
>> xx1=[0:0.1:6]

>> yy1=xx1+2

>> xx2=[0:0.1:6]

>> yy2=3\*ones(1,length(xx2))  
>> plot(xx1,yy1,'r')

>> plot(xx2,yy2,'g')

>> xlabel('x1')

>> ylabel('x2')

>> legend({'x1-x2+2=0','x2-3=0'},'Location', 'northwest')

>> fill([0 0 1],[2 3 3], 'r', 'FaceAlpha', .2)

>> text(0.15, 2.5, 'ОДЗП')

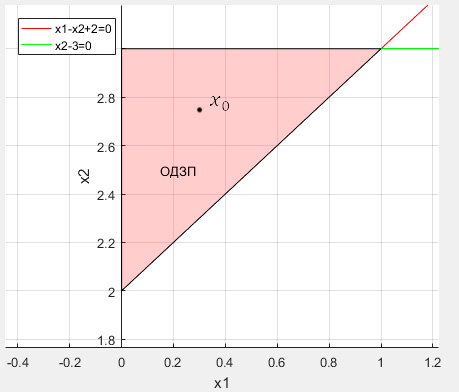


Рисунок 3.3 – Область допустимых значений переменных функции в MATLAB

Выберем начальную точку поиска

## 3.2 Нахождение экстремального значения функции F(x) без учета ограничений на переменные

### *3.2.1 Метод наискорейшего спуска*

Прежде всего найдем составляющие градиента функции:

Градиент функции в точке будет

***1-й шаг.*** Движение осуществляется из точки вдоль вектора в новую точку :

Величину шага найдем из условия:

Отсюда

После подстановки получим:

Градиент функции в точке будет 6488

Найдем значение функции по выражению (3.1) в точке :

***2-й шаг.*** Движение осуществляется из точки вдоль вектора в новую точку :

Величину шага найдем из условия:

Отсюда

После подстановки получим:

Градиент функции в точке будет

Найдем значение функции по выражению (3.1) в точке :

***3-й шаг*** Движение осуществляется из точки вдоль вектора в новую точку :

Величину шага найдем из условия:

Отсюда

После подстановки получим:

Градиент функции в точке будет

Найдем значение функции по выражению (3.1) в точке :

Поскольку норма вектора крайне мала, а данный метод довольно плохо работает в окрестности точки максимума то примем точку максимума как:

А максимальное значение функции .

### 

Рисунок 3.4 – Графическая интерпретация метода наискорейшего спуска

### *3.2.2 Метод Ньютона-Рафсона*

Данный метод дает решение задачи за 1 шаг. Очередная точка поиска вычисляется в соответствии с выражением:

Где *H(x)* – матрица Гессе функции *F(x)*, – обратная по отношению к *H(x)* матрица.

Градиент *F(x)*:

Градиент функции в точке будет

Теперь подставим в изначальную формулу:

Следовательно, в точке функция достигает максимального значения .

### *3.2.3 Нахождение в среде MATLAB*

Найдем экстремум функции в среде MATLAB при помощи команды fminunc. Для этого пишем следующий код:

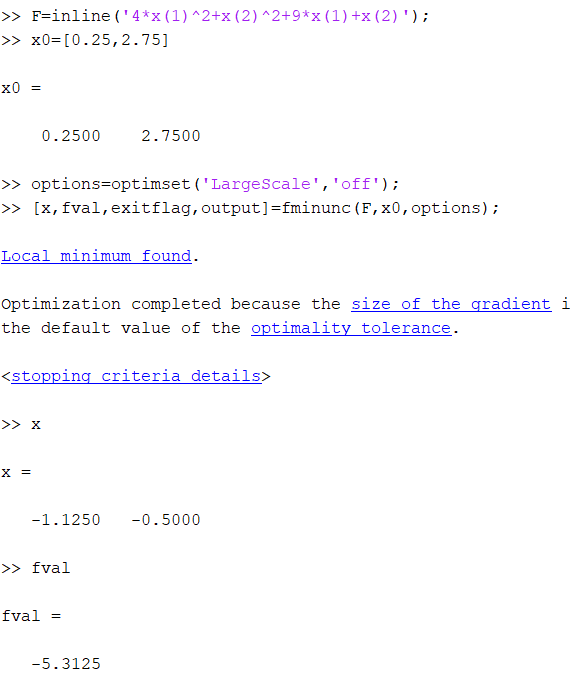
>> F=inline('4\*x(1)^2+x(2)^2+9\*x(1)+x(2)');

>> x0=[0.25, 2.75]

>> options=optimset('LargeScale','off');

>> [x,fval,exitflag,output]=fminunc(F,x0,options);

После выполнения данных команд получим следующий выход:



И получаем, что решение в среде MATLAB полностью совпало с решением метода Ньютона-Рафсона и методом экстремального подъема.

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

### *Метод допустимых направлений Зойтендейка*

Условие задачи:

Градиент функции в точке будет

Движение осуществляется из точки вдоль вектора в новую точку :

Величину шага найдем из условия:

Отсюда

Определим интервал допустимых при котором точка будет находится в ОДЗП, для этого в условие задачи запишем дополнительные ограничения:

Тогда:

Находим величину , которая обеспечит экстремум функции F(x). Воспользуемся уже найденным значением: . Оно не попадает в промежуток. И берем правое значение промежутка т.е. . Значение градиента функции в этой точке:

Градиент функции в точке будет .

Движение выводит за пределы ОДЗП, поэтому очередную точку поиска вычисляет исходя из выражения:

Где новое направление, которое составляет минимальный угол с вектором градиента и направлено либо внутрь, либо по границе ОДЗП. При этом очередная точка должна принадлежать ОДЗП, а функция цели при переходе к очередной точке должна уменьшится максимальным образом.

Найдем направление S1 очередного шага: т.к. *x*1 лежит на оси координат, то проанализировав движение градиента и поиск минимума придём к выводу, что направление очередного шага

При движении из точки в точку следует двигаться по граничной прямой в направлении

Далее рассчитаем координаты точки :

Найдем интервал значения

Тогда

Находим , которое обеспечит максимум функции F(x) в направлении

Значение вне интервала поэтому берем и получаем

Получим точку экстремума

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

Проверим перпендикулярность направления движения и вектора градиента , для этого перемножим эти вектора скалярно:

Найдем направление следующего шага:

Дальнейшего направления не найдено, значит, найденная точка

Обеспечивает минимум функции F(x) с учётом ограничения на переменные:

Найдем значение функции по выражению (3.1) в точке :

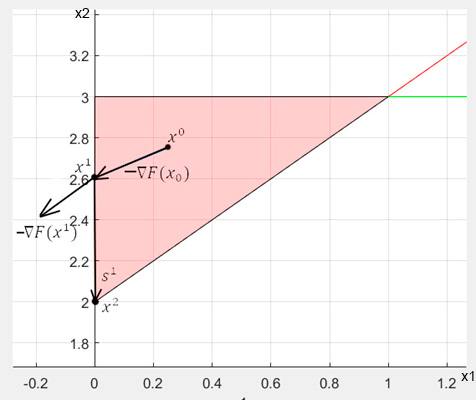


Рисунок 3.5 – Графическая интерпретация метода Зойтендейка

### *Метод линейных комбинаций*

Условие задачи:

Градиент функции в точке будет

Осуществим линеаризацию F(x) относительно точки выражением:

Решается задача ЛП:

|  |  |  |  |
| --- | --- | --- | --- |
| БП | СВЧ | НП | |
|  |  |
|  | -2 | 1 | **-1** |
|  | 3 | 0 | 1 |
|  | 0 | 11 | 6.5 |

Вторая итерация:

|  |  |  |  |
| --- | --- | --- | --- |
| БП | СВЧ | НП | |
|  |  |
|  | 2 | -1 | -1 |
|  | 1 | **1** | 1 |
|  | -13 | 17.5 | 6.5 |

Таблица оптимальна. Делаем корректировку найденного решения в соответствии с выражением:

Определим интервал допустимых значений для при котором точка будет принадлежать ОДЗП.

Поскольку то берем

Поскольку это точка касания одной из линий уровня со стороной ОДЗП, то это точка, при которой достигается минимум функции при наших ограничениях.

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

Вектор градиента направлен в вершине ОДЗП так, что не позволяет двигаться ни внутрь ОДЗП, ни по ее границам.

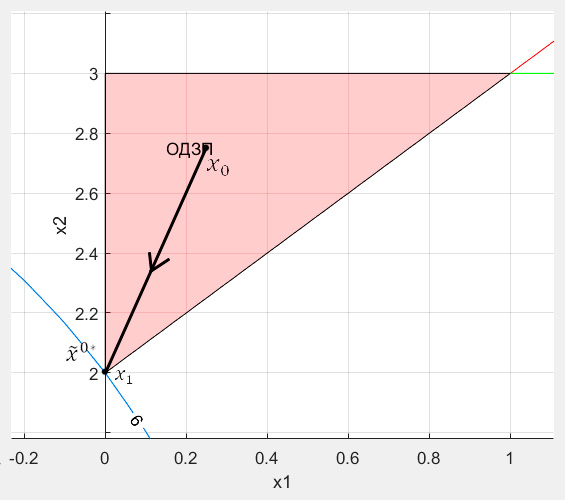


Рисунок 3.6 – Графическая интерпретация метода линейных комбинаций

### *Теорема Куна-Таккера*

Условие задачи:

Составим функцию Лагранжа:

Точка экстремума является седловой точкой с минимумом по x и максимумом по λ, поэтому ограничения приведены к виду :

Условия теоремы Куна-Таккера записываем следующим образом:



Частные производные функции Лагранжа определяются выражениями:

Для того, чтобы вышеуказанные выражения имели вид равенств, введем в них дополнительные переменные:

Решение этой системы можно найти с помощью симплекс процедуры

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| БП | СВЧ | НП | | | |
|  |  |  |  |
|  | 9 | -8 | 0 | -1 | 0 |
|  | 1 | 0 | **-2** | 1 | -1 |
|  | -2 | 1 | -1 | 0 | 0 |
|  | 3 | 0 | 1 | 0 | 0 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| БП | СВЧ | НП | | | |
|  |  |  |  |
|  | 9 | -8 | 0 | -1 | 0 |
|  | -1/2 | 0 | -1/2 | -1/2 | 1/2 |
|  | -5/2 | 1 | -1/2 | **-1/2** | 1/2 |
|  | 7/2 | 0 | 1/2 | 1/2 | -1/2 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| БП | СВЧ | НП | | | |
|  |  |  |  |
|  | 14 | -10 | 1 | -2 | -1 |
|  | 2 | -1 | 0 | -1 | 0 |
|  | 5 | -2 | 1 | -2 | -1 |
|  | 1 | 1 | 0 | 1 | 0 |

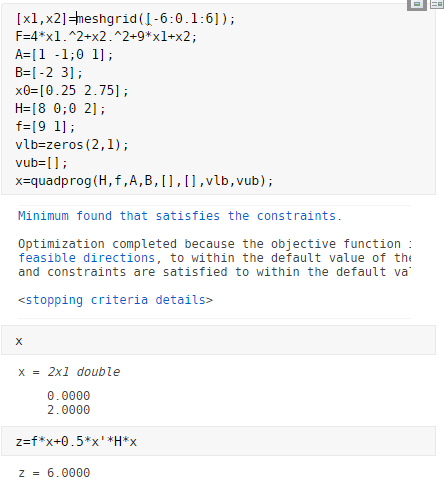
Решение соответствует ,

Кроме того, выполняется условие:

Поэтому является оптимальным.

, что совпадает со всеми предыдущими методами.

### *3.3.4. Квадратичное программирование в MATLAB*



# 

# 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.

# Список использованных источников

[1] Павлова, А. В. Электронный учебно-методический комплекс по учебной дисциплине «Математические основы теории систем» для студентов специальности 1-53 01 07 Информационные технологии и управление в технических системах [Электронный ресурс] / А. В. Павлова, М. К. Хаджинов. – Режим доступа: EUMK\_MOTS\_2013.zip.

[2] Павлова, А. В. Математические основы теории систем: конспект лекций для студентов специальности «Информационные технологии и управление в технических системах». В 2 ч. / А. В. Павлова. – Минск: БГУИР, 2010. – Ч. 2. – 144 с.