Санкт-Петербургский политехнический университет Петра Великого
Институт компьютерных наук и технологий
Кафедра компьютерных систем и программных технологий

КУРСОВАЯ РАБОТА

Дисциплина: **Методы оптимизации**
Тема: **Формулировка и решение задачи выбора оптимального решения с использованием различных математических моделей**

Выполнил студент группы 13541/3 ____________ Д.В. Круминьш
(подпись)

Руководитель ____________ А.Г. Сиднев
(подпись)

Санкт-Петербург
2018 г.

Санкт-Петербургский политехнический университет Петра Великого
Институт компьютерных наук и технологий
Кафедра компьютерных систем и программных технологий

ЗАДАНИЕ
НА ВЫПОЛНЕНИЕ КУРСОВОЙ РАБОТЫ

студенту группы _13541/3_ _____Круминьш_Денис_Валерьевич__________
(номер группы) *(фамилия, имя, отчество)*

1. Тема работы: Формулировка и решение задачи выбора оптимального решения с использованием различных математических моделей

2. Срок сдачи законченного проекта *28.05. 2018*

3. Исходные данные к проекту (работе): Содержательные описания задач поиска оптимальных решений и задачи анализа сетевого графа

4. Содержание пояснительной записки (перечень подлежащих разработке вопросов): введение, формализация и решение многокритериальной оптимизационной задачи, поиск оптимальной стратегии принятия решений с использованием марковских моделей, анализ потокового графа, оптимизация сети систем массового обслуживания, заключение, список использованных источников, приложения.

Дата получения задания: 5 февраля 2018 г.

Руководитель ________________ ___А.Г. Сиднев
(подпись) *(инициалы, фамилия)*

Задание принял к исполнению ____________ ___Д.В. Круминьш_____
(подпись студента) *(инициалы, фамилия)*

5 февраля 2018 г
(дата)

Оглавление

Введение

В данной работе рассматриваются следующие задачи:

1. Формализация многокритериальной оптимизационной задачи, методы сведения к однокритериальной, решение с использованием Optimization Toolbox системы Matlab;
2. Поиск оптимальной стратегии принятия решений с использованием марковских моделей;
3. Решение задачи анализа потокового графа с использованием методики GERT и алгебры потоковых графов;
4. Поиск оптимальных параметров сети систем массового обслуживания.

Глава 1

Формализация многокритериальной оптимизационной задачи, методы сведения к однокритериальной, решение с использованием Optimization Toolbox системы Matlab.

1.1 Постановка задачи

1.1.1 Индивидуальное задание

Задача 2

Фабрика производит два вида изделий, А и Б. Продажа изделий осуществляется на внутреннем рынке и также экспортируется в другие страны. Стоимость изделия А на внутреннем рынке $30, а на внешнем - $45, стоимость изделия Б на внутреннем рынке $20, а на внешнем - $21.

Для изготовления изделий используются два вида станков. Для изготовления 1 единицы изделия А необходима работа первого станка в течении 1 часа и второго – в течение 2 часов, а для изготовления 1 единицы изделия Б необходима работы первого станка в течение 5 часов и второго в течение 7 часов. Ресурс времени непрерывной работы 1 станка 18 часов, а 2 – 20 часов.

Изучение рынка сбыта показало, что спрос на изделие А никогда не превышает 5000 изделий в сутки, а на изделие Б - 9000.

Какое количество изделий каждого вида надо производить в данных условиях для внутреннего и для внешнего рынков, чтобы доход от реализации продукции на внутреннем рынке был максимальным? Какое количество изделий каждого вида надо производить, чтобы минимизировать время использования станков при условии использования не менее 80% ресурса непрерывной работы каждого станка? Какое количество изделий каждого вида надо производить в данных условиях, чтобы доход от реализации продукции на экспорт был максимальным?

1.1.2 Пункты расчетного задания

1. Осуществить переход от многокритериальной задачи к однокритериальной с использованием следующих подходов:
   - Выделение главного критерия;
   - Свертка критериев (аддитивная и мультипликативная);
   - Максимин или минимакс (он же метод максиминной свертки);
   - Метод последовательных уступок;
   - fgoalattain;
   - Ведение метрики в пространстве критериев.
2. Решить задачу стохастического программирования для одной из однокритериальных задач, превратив детерминированное ограничение в вероятностное по схеме

$$P(\sum_{j=1}^{n} a_{ij}x_j - b_i \leq 0) \geq a_i \tag{1.1}$$

Менять a_i в следующем диапазоне $0,1 \leq a_i \leq 0,9$

Считать случайной величиной b_i или элементы $a_{ij}i$-й строки матрицы Aa_{ij} (по выбору).

1.2 Математическая модель задачи многокритериальной оптимизации

- x_{11} – количество изделий А на внутреннем рынке;

- x_{12} – количество изделий А на внешнем рынке;
- x_{21} – количество изделий Б на внутреннем рынке;
- x_{22} – количество изделий Б на внешнем рынке.

Первый критерий. Максимизация дохода на внутреннем рынке

$$f_1(x_{11}, x_{21}) = 30 * x_{11} + 20 * x_{21} \rightarrow max \tag{1.2}$$

Второй критерий. Минимизация использования станков

$$\begin{aligned} f_2(x_{11}, x_{12}, x_{21}, x_{22}) = \\ (x_{11} + x_{12}) + 2 * (x_{11} + x_{12}) + 5 * (x_{21} + x_{22}) + 7 * (x_{21} + x_{22}) = \\ 3 * (x_{11} + x_{12}) + 12 * (x_{21} + x_{22}) \rightarrow min \end{aligned} \tag{1.3}$$

Третий критерий. Максимизация дохода на внешнем рынке

$$f_3(x_{12}, x_{22}) = 45 * x_{12} + 21 * x_{22} \rightarrow max \tag{1.4}$$

Примечание: решений, с ограничением на использование не менее 80% ресурса непрерывной работы станков, найти не удалось. Поэтому данное ограничение было снижено до 70%.
Также часы были переведены в минуты.

Ограничения:
Для станков первого типа

$$1 * (x_{11} + x_{12}) + 5 * (x_{21} + x_{22}) \leq 18 * 60 \tag{1.5}$$

$$1 * (x_{11} + x_{12}) + 5 * (x_{21} + x_{22}) \geq (18 * 60) * 0.7 \tag{1.6}$$

Для станков второго типа

$$2 * (x_{11} + x_{12}) + 7 * (x_{21} + x_{22}) \leq 20 * 60 \tag{1.7}$$

$$2 * (x_{11} + x_{12}) + 7 * (x_{21} + x_{22}) \geq (20 * 60) * 0.7 \tag{1.8}$$

Ограничения по количеству изделий

$$\begin{cases} x_{11} + x_{12} \leq 5000 \\ x_{21} + x_{22} \leq 9000 \end{cases} \tag{1.9}$$

С учетом требований пакета MATLAB к постановке задач оптимизации, необходимо представить целевые функции как поиск минимумов, а ограничения записать в виде

$g(x) \le 0$. Изменим ограничения 1.6 и 1.8, а также введем новые функции, где $z_1 = -f_1, z_3 = -f_3$. В итоге получим:

$$z_1(x_{11}, x_{21}) = -(30 * x_{11} + 20 * x_{21}) \rightarrow min \tag{1.10}$$

$$z_3(x_{12}, x_{22}) = -(45 * x_{12} + 21 * x_{22}) \rightarrow min \tag{1.11}$$

Ограничения:

$$\begin{cases} 1 * (x_{11} + x_{12}) + 5 * (x_{21} + x_{22}) \le 18 * 60 \\ -1 * (x_{11} + x_{12}) - 5 * (x_{21} + x_{22}) \le -(18 * 60) * 0.7 \\ 2 * (x_{11} + x_{12}) + 7 * (x_{21} + x_{22}) \le 20 * 60 \\ -2 * (x_{11} + x_{12}) - 7 * (x_{21} + x_{22}) \le -(20 * 60) * 0.7 \\ x_{11} + x_{12} \le 5000 \\ x_{21} + x_{22} \le 9000 \end{cases} \tag{1.12}$$

1.2.1 Поиск оптимумов частных критериев

Найдем оптимумы каждой из целевых функций независимо от других. Для этого необходимо решить три задачи однокритериальной оптимизации: для z_1, f_2, z_3 при тех же ограничениях на $x_{11}, x_{12}, x_{21}, x_{22}$ что имеют место для задачи многокритериальной оптимизации.

Для решение данной задачи, был использован MATLAB, код с общими параметрами приведен в приложении 1.

```
%% Поиск оптимумов частных критериев
startingPoint = lb;
[x, z1_opt] = fmincon(z1, startingPoint, A, b, Aeq, beq, lb, ub)
[x, f2_opt] = fmincon(f2, startingPoint, A, b, Aeq, beq, lb, ub)
[x, z3_opt] = fmincon(z3, startingPoint, A, b, Aeq, beq, lb, ub)
```

Листинг 1.1: Поиск оптимумов частных критериев

```
x =
  236.0000
    0.0000
  104.0000
    0.0000
z1_opt =
  -9.1600e+03

x =
```

```
     0.0000
     0.0000
    75.6000
    75.6000
f2_opt =
    1.8144e+03

x =
   0.0000
   236.0000
   0.0000
   104.0000

z3_opt =
   −1.2804e+04
```

Листинг 1.2: Результаты выполнения листинга 1.1

Таким образом, были получены следующие оптимальные значения:

$$z_1^{min} = -9160, f_2^{min} = 1814.4, z_3^{min} = -12804 \tag{1.13}$$

Откуда следует:

1. $f_1^{max} = 9160\$$ - максимальный доход на внутреннем рынке;
   - x_{11} - 236;
   - x_{21} - 104.
2. $f_2^{min} = 1814.4$ часов - общее время использования первого и второго станка;
   - x_{21} - 75.6;
   - x_{22} - 75.6;
   - Примечание: значение в 151.2(75.6+75.6) изделий Б, упирается в нижнюю границу(756 часов(70%)) первого станка.
3. $f_3^{max} = 12804\$$ - максимальный доход на внешнем рынке.
   - x_{12} - 236;
   - x_{22} - 104.

Также были посчитаны значения прочих критериев при оптимуме каждого критерия.

x_{11}	x_{12}	x_{21}	x_{22}	f_1	f_2	f_3
Оптимум f_1						

236	0	104	0	9160	1956	0
Оптимум f_2						
0	0	75.6	75.6	1512	1814.4	1587.6
Оптимум f_3						
0	236	0	104	0	1956	12804

Таблица 1.1: Оптимумы критериев и значения функций

Как видно из таблицы, оптимум первого или третьего критерия означает 0 велечину другого критерия.

1.3 Переход от многокритериальной задачи к однокритериальной

1.3.1 Выделение главного критерия

Один из критериев - главный - имеет существенно более высокий приоритет, чем все остальные, но по остальным критериям вариант тоже не должен быть слишком плох. Пусть главный критерий - первый, следовательно, для оставшихся целевых функций необходимо указать нижние границы. Предположим, что общее время изготовления продукции (f_2) должно быть не более 1950 часов, а доход на внешнем рынке должен быть больше 1000$. Таким образом, к задаче добавляются еще 2 ограничения:

$$3 * (x_{11} + x_{12}) + 12 * (x_{21} + x_{22}) \leq 1950 \tag{1.14}$$

$$45 * x_{12} + 21 * x_{22} \geq 1000 \tag{1.15}$$

В соответствии с изменениями скрипт был дополнен ограничениями(Код приведен в приложении 2).

После выполнения программы были получены следующие результаты:

```
x =

  203.7778
   22.2222
  106.0000
    0.0000
```

```
f_val =

  -8.2333e+03
```
Листинг 1.3: Результаты выполнения листинга 4.11

Таким образом было получено значение в 8233.3$, что составляет 89% от оптимума. Время использования станков - 1950 часов(упор в ограничение), доход на внешнем рынке - 1000$(упор в ограничение).

- f_1 - 8233.3$ (89.9% от оптимума);
- f_2 - 1950 часов (на 7.4% больше оптимума);
- f_3 - 1000$ (7.8% от оптимума);
- 9 233.3$ - общий доход.

Столь низкий процент от оптимума у f_3 объясняется тем, что большая часть изделий продавалась на внутренний рынок, а не на внешний.

1.3.2 Свертка критериев

Аддитивная свертка критериев

Для использования метода аддитивной свертки необходимо выполнить нормировку критериев, с тем чтобы сделать их значения соизмеримыми, а единицы измерения – безразмерными. Выполним нормировку следующим образом:

$$\overline{z_1} = \frac{z_1}{|z_1^{min}|} = -\frac{30 * x_{11} + 20 * x_{21}}{9160} = -\frac{3 * x_{11} + 2 * x_{21}}{916} \tag{1.16}$$

$$\overline{f_2} = \frac{f_2}{|f_2^{min}|} = \frac{3 * (x_{11} + x_{12}) + 12 * (x_{21} + x_{22})}{1814.4} = \frac{x_{11} + x_{12} + 4 * (x_{21} + x_{22})}{604.8} \tag{1.17}$$

$$\overline{z_3} = \frac{z_3}{|z_3^{min}|} = \frac{-45 * x_{12} + 21 * x_{22}}{12804} = \frac{-15 * x_{12} + 7 * x_{22}}{4266.8} \tag{1.18}$$

Формула аддитивной свертки имеет вид:

$$F(x) = \sum_{i=1}^{r} \lambda_i f_i(x), 0 < \lambda_i < 1, \sum_i \lambda_i = 1, \tag{1.19}$$

где $f_i(x)$ - критерии оптимальности, r – их общее число, а λ_i - параметры важности. Примем $\lambda_1 = 0.4, \lambda_2 = 0.2, \lambda_3 = 0.4$. Для этого добавим к листингу 1.1 следующий код:

```
% Аддитивная свертка
z1_norm = @(N) z1(N)/abs(z1_opt);
f2_norm = @(N) f2(N)/abs(f2_opt);
z3_norm = @(N) z3(N)/abs(z3_opt);
f = @(N) 0.4*z1_norm(N) +0.2*f2_norm(N) + 0.4*z3_norm(N);
A = [1,1,5,5;
     -1,-1,-5,-5;
     2,2,7,7;
     -2,-2,-7,-7;
     1,1,0,0;
     0,0,1,1
     3,3,12,12;
     0,-45,0,-21];
b = [Tmax1; -Tmin1; Tmax2; -Tmin2;  5000; 9000; 1950; -1000];

[N, f_opt] = fmincon(f, startingPoint, A, b, Aeq, beq, lb, ub)
```

Листинг 1.4: Аддитивная свертка

```
N =
     0.0009
   225.9990
   105.9997
     0.0004

f_opt =
   -0.1953

>> f1(N)
ans =
    2.1200e+03

>> f2(N)
ans =
    1.9500e+03

>> f3(N)
ans =
    1.0170e+04
```

Листинг 1.5: Результаты выполнения листинга 1.4

Метод аддитивной свертки позволил получить решение:

- f_1 - 2120$ (23.1% от оптимума);
- f_2 - 1950 часов (на 7.4% больше оптимума);
- f_3 - 10170$ (79.4% от оптимума);
- 12 290$ - общий доход.

Мультипликативная свертка критериев

Формула мультипликативной свертки имеет вид:

$$F(x) = \prod_{i=1}^{r} f_i(x)^{\lambda_i} \tag{1.20}$$

где $f_i(x)$ - критерии оптимальности, r - их общее число, а λ_i - показатели важности. Примем $\lambda_1 = 0.4, \lambda_2 = 0.2, \lambda_3 = 0.4$. Данные значения были выбраны из соображения того что доход более важен чем минимизация времени станков. А также что доход на внутреннем и внешнем рынке одинаково важны. Нормировка уже была произведена в аддитивной свертки, в итоге получим следующую задачу однокритериальной оптимизации:

$$f = \overline{z_1}^{0.4} * \overline{f_2}^{0.2} * \overline{z_3}^{0.4} \tag{1.21}$$

Для этого добавим к листингу 1.1 следующий код:

```
% Мультипликативная свертка
startingPoint = lb;
z1_norm = @(N) z1(N)/abs(z1_opt);
f2_norm = @(N) f2(N)/abs(f2_opt);
z3_norm = @(N) z3(N)/abs(z3_opt);
f = @(N) (-(f1(N)/9160)^0.4)*((f2(N)/1814.4)^0.2)*((f3(N)/12804)^0.4)
A = [1,1,5,5;
     -1,-1,-5,-5;
     2,2,7,7;
     -2,-2,-7,-7;
     1,1,0,0;
     0,0,1,1
     3,3,12,12;
     0,-45,0,-21];
b = [Tmax1; -Tmin1; Tmax2; -Tmin2;  5000; 9000; 1950; -1000];

[N, f_opt] = fmincon(f, startingPoint, A, b, Aeq, beq, lb, ub)
```

Листинг 1.6: Мультипликативная свертка

```
N =
    77.6665
   148.3330
   105.9988
     0.0014

 f_opt =
    −0.5858

>> f1(N)
ans =
    4.4500e+03

>> f2(N)
ans =
    1.9500e+03

>> f3(N)
ans =
    6.6750e+03
```

Листинг 1.7: Результаты выполнения листинга 1.6

Метод мультипликативной свертки позволил получить решение:

- f_1 - 4450$ (48.6% от оптимума);
- f_2 - 1950 часов (на 7.4% больше оптимума);
- f_3 - 6675$ (52.1% от оптимума);
- 11 125$ - общий доход.

Мультипликативная свертка позволила получить более компромиссное решение для f_1 и f_3, однако больший общий доход позволила получить аддитивная свертка.

1.3.3 Минимакс (максимин)

Максиминную свертку представим в следующем виде: $C_i(a) = \min w_i C_i(a)$

Решение a^* является наилучшим, если для всех a выполняется условие $C(a^*) \geq C(a)$, или $a^* = \arg\max C(a) = \arg\max\min w_i C_i(a)$.

Для реализации максиминной свертки необходимо в fminimax передавать функции обратные целевым (функция funminmax). Так как оцениваемые показатели разновелики, необходимо нормировать критерии. Что было произведено ранее.

Решение задачи представлено как скрипт в среде Matlab, для этого листинг 4.10 был дополнен:

```
%Минимакс — максимин
startingPoint = lb;
[x, f] = fminimax (@funminmax , startingPoint , A , b, Aeq, beq, lb , ub )

function f = funminmax (X)
f(1) = -(30*X(1) - 20*X(3))/9160;
f(2) = -(3*(X(1)+X(2))+12*(X(3)+X(4)))/1814.4;
f(3) = -(45*X(2) - 21*X(4))/12804;
end
```

Листинг 1.8: Минимакс (максимин)

```
x =

  123.4525
  112.5475
   46.9046
   57.0954


f =
   -0.3019    -1.0780    -0.3019

>> f1(x)
ans =
   4.6417e+03

>> f2(x)
ans =
        1956

>> f3(x)
ans =
   6.2636e+03
```

Листинг 1.9: Результаты выполнения листинга 1.8

Метод минимакс (максимин) позволил получить решение:

- f_1 - 4641.7$ (50.6% от оптимума);
- f_2 - 1956 часов (на 7.8% больше оптимума);
- f_3 - 6263.6$ (48.9% от оптимума);

- 10905.3$ - общий доход.

Процентное соотношение первого и второго критерия относительно оптимума примерно равное, второй критерий по сути игнорируется.

1.3.4 Метод последовательных уступок

Для решения данной задачи была выбрана уступка = 10%. Предположим, что критерии пронумерованы в следующем порядке важности:

$$z_1 > f_2 > z_3$$

Для первого критерия уже решена задача поиска оптимального значения в п 1.2.1. То есть:

$$9160 * 0.9 = 8244$$

То ограничения критерия выглядит следующим образом:

$$-30 * x_{11} - 20 * x_{21} \leq -8244$$

Запишем ограничения в скрипт

```
...
% Функциональные ограничения
A = [1 ,1 ,5 ,5;
     -1,-1,-5,-5;
     2 ,2 ,7 ,7;
     -2,-2,-7,-7;
     1 ,1 ,0 ,0;
     0 ,0 ,1 ,1;
     -30,0,-20,0];
b = [Tmax1; -Tmin1; Tmax2; -Tmin2;  5000; 9000; -8244];

...

[x, z3_opt] = fmincon(z3, startingPoint, A, b, Aeq, beq, lb, ub)
```

Листинг 1.10: Последовательные уступки

```
x =
  205.4667
   30.5333
  104.0000
    0.0000
```

```
z3_opt =
  −1.3740e+03

>> f1(x)
ans =
    8.2440e+03
```

Листинг 1.11: Результат выполнения

Как и ожидалось при минимизации функции z_3 учитывалось и ограничения для z_1, что существенное уменьшило результат для z_3.

В соответствии с полученным значением введем ограничение для второго критерия.

$$1374 * 0.9 = 1236.6$$

Ограничения критерия выглядит следующим образом:

$$-45 * x_{12} - 21 * x_{22} \leq -1236.6$$

Запишем ограничения в скрипт

```
...
% Функциональные ограничения
A = [1,1,5,5;
     −1,−1,−5,−5;
     2,2,7,7;
     −2,−2,−7,−7;
     1,1,0,0;
     0,0,1,1;
     −30,0,−20,0;
     0,−45,0,−21];
b = [Tmax1; −Tmin1; Tmax2; −Tmin2;  5000; 9000; −8244; −1236.6];

...

[x, z3_opt] = fmincon(z3, startingPoint, A, b, Aeq, beq, lb, ub)
```

Листинг 1.12: Последовательные уступки

```
x =
  204.9969
   27.4800
  104.7046
    0.0000

f2_opt =
    1.9539e+03
```

```
>> f1(x)
ans =
    8.2440e+03

>> f3(x)
ans =
    1.2366e+03
```

Листинг 1.13: Результат выполнения

Итоговые результаты:

- f_1 - 8244$ (90% от оптимума);
- f_2 - 1953.9 часов (на 7.6% больше оптимума);
- f_3 - 1236.6$ (9.6% от оптимума);
- 9 480.6$ - общий доход.

По результатам видно, что за счет последовательности, при оптимизации каждого следующего критерия учитываются и уступка предыдущих критериев.

1.3.5 Метод достижения цели (fgoalattain)

Fgoalattain решает задачу достижения цели, которая является одной из формулировок задач для векторной оптимизации. x = fgoalattain(fun, x_0, goal, weight, ...):

- fun – целевая функция,
- x_0 – начальные значения,
- goal – целевые значения,
- weight – веса.

Пусть goal = ($z_1^{min}, f_2^{min}, z_3^{min}$), w = ($|z_1^{min}|, |f_2^{min}|, |z_3^{min}|$). Тогда скрипт для решения задачи будет выглядеть следующим образом:

```
...
%% fgoalattain
f = @(N) [z1(N), f2(N), z3(N)];
goal = [z1_opt, f2_opt, z3_opt];
w = abs(goal);
A = [1,1,5,5;
     -1,-1,-5,-5;
```

```
     2,2,7,7;
     −2,−2,−7,−7;
     1,1,0,0;
     0,0,1,1];
b = [Tmax1; −Tmin1; Tmax2; −Tmin2;  5000; 9000];
lb = [0; 0; 0; 0];
ub = [5000; 5000; 9000; 9000];
startingPoint = lb;

[N, f_opt, af] = fgoalattain(f, startingPoint, goal, w, A, b, Aeq, beq, lb, ub)
```

Листинг 1.14: Метод достижения цели

```
N =
  115.2789
  120.7211
   56.8321
   47.1679



f_opt =
   1.0e+03 *
   −4.5950     1.9560    −6.4230

af =
    0.4984
```

Листинг 1.15: Результат выполнения

Значение переменной af, говорит о том, что полученное решение на 49.8% хуже цели.

Итоговые результаты:

- f_1 - 4595$ (50.2% от оптимума);
- f_2 - 1956 часов (на 7.8% больше оптимума);
- f_3 - 6423$ (50.2% от оптимума);
- 11 018$ - общий доход.

Оба критерия f_1 и f_3 составляют одинаковую долю от своих оптимальных значений, в то время как второй критерий был проигнорирован, отдавая больший приоритет двум другим критериям.

1.3.6 Введение метрики в пространстве критериев

Для перехода к однокритериальной задаче оптимизации методом введения метрики в пространстве целевых функций необходимо определить координаты «идеальной» точки $a = (f_1^*, f_2^*, ..., f_r^*)$, где $f_i = min f_i(x)$. Эти значения уже были получены в п. 1.2.1, и поэтому:

$$a = (-9160, 1814.4, -12804)$$

Введем в пространстве критериев метрику в виде евклидова расстояния:

$$p(y, a) = [\sum_{i=1}^{r}(a_i - y_i)^2]^{\frac{1}{2}} \tag{1.22}$$

Тогда за целевую функцию (обобщенный критерий), с учётом необходимости нормировки, можно взять выражение:

$$f = \sum_{i=1}^{r}(\frac{a_i - f_i}{f_i^*})^2 = \sum_{i=1}^{r}(1 - \frac{f_i}{f_i^*})^2 \tag{1.23}$$

Таким образом, получаем следующую задачу оптимизации:

$$f = (1 - \frac{z_1}{z_1^{min}})^2 + (1 - \frac{f_2}{f_2^{min}})^2 + (1 - \frac{z_3}{z_3^{min}})^2 \tag{1.24}$$

```
...
% Параметрические ограничения
lb = [0; 0; 0; 0];
ub = [5000; 5000; 9000; 9000];

%% Поиск оптимумов частных критериев
startingPoint = lb;
[x, z1_opt] = fmincon(z1, startingPoint, A, b, Aeq, beq, lb, ub)
[x, f2_opt] = fmincon(f2, startingPoint, A, b, Aeq, beq, lb, ub)
[x, z3_opt] = fmincon(z3, startingPoint, A, b, Aeq, beq, lb, ub)

%% Введение метрики в пространстве критериев
f = @(N) (1-z1(N)/z1_opt)^2+(1-f2(N)/f2_opt)^2+(1-z3(N)/z3_opt)^2;
[N, f_opt] = fmincon(f, startingPoint, A, b, Aeq, beq, lb, ub)
```

Листинг 1.16: Метод достижения цели

```
N =
    83.7126
   152.2874
   103.9999
     0.0001
```

```
f_opt =
    0.4709

>> f1(N)
ans =
   4.5914e+03

>> f2(N)
ans =
   1.9560e+03

>> f3(N)
ans =
   6.8529e+03
```

Листинг 1.17: Результат выполнения

Итоговые результаты:

- f_1 - 4591.4$ (50.1% от оптимума);
- f_2 - 1956 часов (на 7.8% больше оптимума);
- f_3 - 6852.9$ (53.5% от оптимума);
- 11 444.3$ - общий доход.

1.4 Оценка Парето-оптимальности полученных решений

Для того чтобы уменьшить количество альтернатив, среди которых лицо, принимающее решение (ЛПР), должно сделать выбор, можно выделить множество Парето среди всех полученных решений. Для этого была составлена таблица и построен график.

Столбец **SUM** означает общий доход с внешнего и внутреннего рынка, то есть $f_1 + f_3$.

Метод перехода к однокритериальной задаче	x_{11}	x_{12}	x_{21}	x_{22}	f_1	f_2	f_3	**SUM**
A. Выделение главного критерия	203.8	22.2	106	0	8233.3	1950	1000	9233.3

Б1. Аддитивная свертка	0	226	106	0	2120	1950	10170	12290
Б2. Мультипликативная свертка	77.7	148.3	106	0	4450	1950	6675	11125
В. Минимакс	123.45	112.55	46.9	57.1	4641.7	1956	6263.6	10905.3
Г. Метод последовательных уступок	205	27.5	105	0	8244	1953.9	1236.6	9480.6
Д. Метод достижения цели (fgoalattain)	115.3	120.7	56.8	47.2	4595	1956	6423	11018
Е. Введение метрики в пространстве критериев	83.7	152.3	104	0	4591.4	1956	6852.9	11444.3

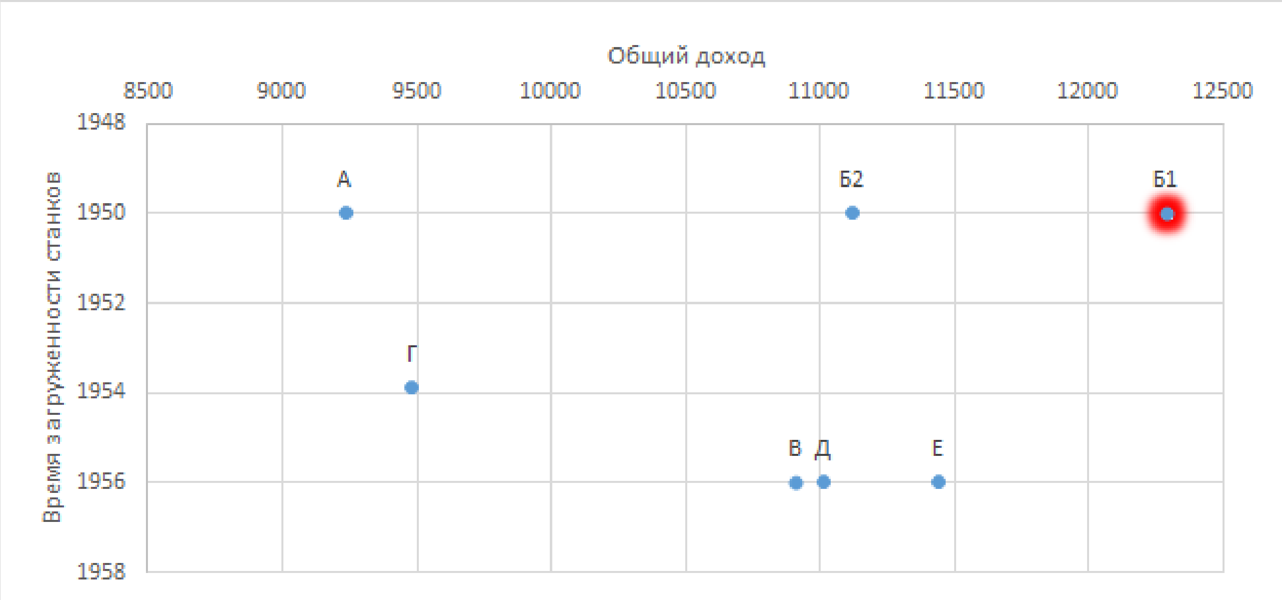

Рис. 1.1: Оценки от полученных решений на плоскости критериев: красным выделено множество Парето

Парето-оптимальными являются только решение Б1. С точки зрения ЛПР оно также является лучшим, так как позволяет достичь максимального дохода.

1.5 Решение задачи стохастического программирования

Рассмотрим задачу стохастического программирования на основе задачи однокритериальной оптимизации, которая была получена из исходной методом введения метрики в пространстве критериев (п. 1.3.6).

Преобразуем третье ограничение системы:

$$2x_{11} + 2x_{12} + 7 * x_{21} + 7 * x_{22} \leq 1200$$

в вероятностное, тогда:

$$P(\alpha_{11}x_{11} + \alpha_{12}x_{12} + \alpha_{21}x_{21} + \alpha_{22}x_{22} \leq 1200) \geq \alpha$$

где все a_{ij} нормально распределены и имеют следующие математические ожидания и дисперсии:

$$M(\alpha_{11}) = 2, M(\alpha_{12}) = 2, M(\alpha_{21}) = 7, M(\alpha_{22}) = 7$$
$$D(\alpha_{11}) = 1, D(\alpha_{12}) = 1, D(\alpha_{21}) = 3.5, D(\alpha_{22}) = 3.5$$

Где СКО равно половине математического ожидания. По таблице функции распределения стандартного нормального закона находим $K_a(0.1 \leq \alpha \leq 0.9)$

$$K_{0.1} = -1.2816, K_{0.2} = -0.8416, K_{0.3} = -0.5244, K_{0.4} = -0.2533, K_{0.5} = 0$$
$$K_{0.6} = 0.2533, K_{0.7} = 0.5244, K_{0.8} = 0.8416, K_{0.9} = 1.2816$$

Таким образом, вероятностное ограничение становится эквивалентно детерминированному неравенству:

$$2x_{11} + 2x_{12} + 7 * x_{21} + 7 * x_{22} + K_\alpha * \sqrt{x_{11}^2 + x_{12}^2 + 3.5 * x_{21}^2 + 3.5 * x_{22}^2} \leq 1200 \quad (1.25)$$

Решение задачи представлено как скрипт в программе Matlab, код представлен в приложении 3.

```
alpha = 0.1, X_11 = 271.4401, X_12 = 278.9110, x_21 = 41.1291, x_22 =  0.001,
    ↪ f1 = 8965.78, f2 = 2145, f3 = 12551.01
alpha = 0.2, X_11 = 271.4492, X_12 = 278.9046, x_21 = 41.1151, x_22 =  0.014,
    ↪ f1 = 8965.78, f2 = 2145, f3 = 12551.01
alpha = 0.3, X_11 = 271.4494, X_12 = 278.9048, x_21 = 41.1172, x_22 =  0.012,
    ↪ f1 = 8965.83, f2 = 2145, f3 = 12550.97
alpha = 0.4, X_11 = 155.5649, X_12 = 205.9867, x_21 = 78.8894, x_22 =  0.000,
    ↪ f1 = 6244.73, f2 = 2031, f3 = 9269.41
alpha = 0.5, X_11 = 83.7126, X_12 = 152.2874, x_21 = 103.9999, x_22 =  0.000,
    ↪ f1 = 4591.37, f2 = 1956, f3 = 6852.93
alpha = 0.6, X_11 = 63.7924, X_12 = 88.9527, x_21 = 74.5707, x_22 = 46.080, f1
    ↪ = 3405.19, f2 = 1906, f3 = 4970.56
```

```
alpha = 0.7, X_11 = 27.7783, X_12 = 43.1165, x_21 = 73.8482, x_22 = 63.173, f1
    ↪ = 2310.31, f2 = 1857, f3 = 3266.87
alpha = 0.8, X_11 = 0.0000, X_12 = 0.0000, x_21 = 74.3433, x_22 = 74.343, f1 =
    ↪ 1486.87, f2 = 1784, f3 = 1561.21
alpha = 0.9, X_11 = 0.0000, X_12 = 0.0000, x_21 = 70.9449, x_22 = 70.990, f1 =
    ↪ 1418.90, f2 = 1703, f3 = 1490.78Для
 детерминированных ограничений:
X_11 = 83.7126, X_12 = 152.2874, x_21 = 103.9999, x_22 =  0.000, f1 = 4591.37,
    ↪ f2 = 1956, f3 = 6852.93
```

Листинг 1.18: Результат выполнения

α	x_{11}	x_{12}	x_{21}	x_{22}	f_1	f_2	f_3
0.1	271.4401	278.9110	41.1291	0.001	8965.78	2145	12551.01
0.2	271.4492	278.9046	41.1151	0.014	8965.78	2145	12551.01
0.3	271.4494	278.9048	41.1172	0.012	8965.83	2145	12550.97
0.4	155.5649	205.9867	78.8894	0.000	6244.73	2031	9269.41
0.5	83.7126	152.2874	103.9999	0.000	4591.37	1956	6852.93
0.6	63.7924	88.9527	74.5707	46.080	3405.19	1906	4970.56
0.7	27.7783	43.1165	73.8482	63.173	2310.31	1857	3266.87
0.8	0.0000	0.0000	74.3433	74.343	1486.87	1784	1561.21
0.9	0.0000	0.0000	70.9449	70.990	1418.90	1703	1490.78
Решение задачи при детерминированных ограничениях							
-	83.7126	152.2874	103.9999	0.000	4591.37	1956	6852.93

Задача чувствительна к выбранному ограничению, т.к. для различных К получились разные результаты.

Увеличение доверительной вероятности α приводит к ухудшению оценок получаемых решений по первому и второму критерию. При $\alpha = 0.5, K_\alpha = 0$в случае чего получается решение, совпадающее с решением задачи с детерминированными ограничениями.

При $\alpha < 0.5$ квантиль приобретает отрицательное значение. За счет чего ограничения **ослабевают**(их выполнение становится менее важным), и например решение при $\alpha = 0.1$ практические совпадает с оптимум для первого и второго критерия одновременно, чего не удалось добиться в каких-либо методах оптимизации.

А при $\alpha > 0.5$ ограничения становяться более **жесткими**, что уменьшает общий доход.

Глава 2

Поиск оптимальной стратегии принятия решений с использованием марковских моделей

2.1 Постановка задачи

Вариант: 12, решить задачу методом итераций по стратегиям для $N = \infty$

Крупная фирма, производящая моющие средства и пользующаяся широкой известностью в связи с успехами в исследованиях по созданию новых продуктов и их рекламированию, выпустила на рынок новый высококачественный стиральный порошок, названный LYE. Руководитель, возглавляющий производство этого продукта, совместно с отделом рекламы разрабатывает специальную рекламную кампанию по сбыту порошка, для которой принят девиз «Порошок LYE нужен всем!» Как и все продукты фирмы, новый продукт в течение первого полугодия будет иметь высокий уровень сбыта. Руководитель полагает, что с вероятностью 0,8 этот уровень сбыта сохранится и в последующем полугодии при условии проведения особой рекламной кампании и что эта вероятность составит всего 0,5, если такую кампанию не проводить. В случае, если уровень сбыта снизится до среднего, у руководителя имеются две возможности. Он может дать указание о проведении исследований с целью улучшения качества продукта. При этом условии с вероятностью 0,7 уровень сбыта к началу следующего полугодия повысится до первоначального высокого значения. С другой стороны, можно ничего не предпринимать в отношении улучшения качества продукта. Тогда с вероятностью 0,6 в начале последующего полугодия уровень сбыта останется средним, однако вследствие изменений потребительских вкусов он может вновь подняться до высокого значения лишь с вероятностью 0,4.

Если сбыт нового стирального порошка начинается на высоком уровне при обычной рекламе, то прибыли в течение полугодия равны 19 единицам в случае, когда этот уровень сохраняется, и равны 13, если уровень сбыта падает. При проведении специальной рекламной кампании соответствующие показатели равны 4,5 и 2 единицам. Если начальный уровень сбыта окажется средним и при этом проводятся исследования с целью улучшения качества продукции, то прибыли составят 11 единиц в случае, когда уровень сбыта поднимается до высокого, и 9 единиц в противном случае. При сохранении продукта в неизменном виде соответствующие прибыли равны 13 и 3 единицам. Предположим, что одна и та же проблема принятия решений относительно сбыта стирального порошка LYE повторяется через каждые полгода в течение бесконечного планового периода.

2.2 Метод итераций по стратегиям

Для начала выпишем все известные параметры задачи.

Система может быть в двух состояних:

1. хороший сбыт(S_1);
2. средний сбыт(S_2).

Организация может предпринять следующие действия(далее стратегии):

1. всегда улучшать сбыт(X_1)
   - при S_1 - создание специальной рекламы(D_1);
   - при S_2 - проведение исследований(D_2).
2. улучшать сбыт только при хорошем сбыте(X_2)
   - при S_1 - создание специальной рекламы(D_1);
   - при S_2 - ничего не делать(D_3).
3. улучшать сбыт только при среднем сбыте(X_3)
   - при S_1 - ничего не делать(D_3);
   - при S_2 - проведение исследований(D_2).

4. всегда ничего не делать(X_4)
   - при S_1 - ничего не делать(D_3);
   - при S_2 - ничего не делать(D_3).

На основе данной информации составим матрицы переходных вероятностей P_1, P_2, P_3, P_4 соответсвующие стратегиям X_1, X_2, X_3, X_4.

$$P_1 = \begin{pmatrix} 0.8 & 0.2 \\ 0.7 & 0.3 \end{pmatrix} P_2 = \begin{pmatrix} 0.8 & 0.2 \\ 0.4 & 0.6 \end{pmatrix} P_3 = \begin{pmatrix} 0.5 & 0.5 \\ 0.7 & 0.3 \end{pmatrix} P_4 = \begin{pmatrix} 0.5 & 0.5 \\ 0.4 & 0.6 \end{pmatrix}$$

Также составим матрицы доходов R_1, R_2, R_3, R_4.

$$R_1 = \begin{pmatrix} 4.5 & 2 \\ 11 & 9 \end{pmatrix} R_2 = \begin{pmatrix} 4.5 & 2 \\ 13 & 3 \end{pmatrix} R_3 = \begin{pmatrix} 19 & 13 \\ 11 & 9 \end{pmatrix} R_4 = \begin{pmatrix} 19 & 13 \\ 13 & 3 \end{pmatrix}$$

Множество допустимых стратегий $G = \{X_1, X_2, X_3, X_4\}$.

2.2.1 Этап(1) оценивания параметров

Выбираем стратегию τ - X_4. Тогда, матрицы переходных вероятностей и доходов будут следующими:

$$P = \begin{pmatrix} 0.5 & 0.5 \\ 0.4 & 0.6 \end{pmatrix} R = \begin{pmatrix} 19 & 13 \\ 13 & 3 \end{pmatrix}$$

Учитывая, что $F_\tau(2) = 0$, получаем систему линейных алгебраических уравнений:

$$\begin{cases} E_\tau - (1 - 0.5) * F_\tau(1) = 16 \\ E_\tau - (1 - 0.4) * F_\tau(1) = 7 \end{cases}$$

```
syms Et Ft
eqn1 = Et - (1-0.5)*Ft == 16;
eqn2 = Et - (1-0.4)*Ft == 7;
[A, B] = equationsToMatrix([eqn1, eqn2], [Et, Ft])
X = linsolve(A, B)
```

Листинг 2.1: Скрпит для решения системы уравнений

В результате выполнения скрипта matlab, было получено единственное решение:

$$E_\tau = 61; F_\tau(1) = 90$$

2.2.2 Этап(1) улучшения стратегии

Для каждого состояния S_j, где j от 1 до m, найдем допустимое решение, на котором достигается:

$$max(v_j(X_i) + \sum_{k=1}^{m} p_{jk}(X_i)F_\tau(k)$$

S_j	$\varphi_i = v_j(X_i) + p_{j1}(X_i)F_i(1)$				$max\varphi_i$	X_j
	i=1	i=2	i=3	i=4		
1	4+0.8*(90)= 76	4+0.8*(90)= 76	16+0.5*(90)= 61	16+0.5*(90)= 61	76	X_1, X_2
2	10.4+0.7*(90)= 73.4	7+0.4*(90)= 40	10.4+0.7*(90)= 73.4	7+0.4*(90)= 40	73.4	X_1, X_3

2.2.3 Этап(2) оценивания параметров

Выбираем стратегию τ - X_3. Тогда, матрицы переходных вероятностей и доходов будут следующими:

$$P = \begin{pmatrix} 0.5 & 0.5 \\ 0.7 & 0.3 \end{pmatrix} R = \begin{pmatrix} 19 & 13 \\ 11 & 9 \end{pmatrix}$$

Учитывая, что $F_\tau(2) = 0$, получаем систему линейных алгебраических уравнений:

$$\begin{cases} E_\tau - (1 - 0.5) * F_\tau(1) = 16 \\ E_\tau - (1 - 0.7) * F_\tau(1) = 10.4 \end{cases}$$

```
syms Et Ft
eqn1 = Et - (1-0.5)*Ft == 16;
eqn2 = Et - (1-0.7)*Ft == 10.4;
[A, B] = equationsToMatrix([eqn1, eqn2], [Et, Ft])
X = linsolve(A, B)
```

Листинг 2.2: Скрпит для решения системы уравнений

В результате выполнения скрипта matlab, было получено единственное решение:

$$E_\tau = 2; F_\tau(1) = -28$$

2.2.4 Этап(2) улучшения стратегии

Для каждого состояния S_j, где j от 1 до m, найдем допустимое решение, на котором достигается:

$$max(v_j(X_i) + \sum_{k=1}^{m} p_{jk}(X_i)F_\tau(k)$$

S_j	$\varphi_i = v_j(X_i) + p_{j1}(X_i)F_i(1)$				$max\varphi_i$	X_j
	i=1	i=2	i=3	i=4		
1	4+0.8*(-28)= -18.4	4+0.8*(-28)= -18.4	16+0.5*(-28)= 2	16+0.5*(-28)= 2	2	X_3, X_4
2	10.4+0.7*(-28)= -9.2	7+0.4*(-28)= -4.2	10.4+0.7*(-28)= -9.2	7+0.4*(-28)= -4.2	-4.2	X_2, X_4

Так как $t \neq \tau$, то снова переходим к этапу оценивания параметров.

2.2.5 Этап(3) оценивания параметров

Выбираем стратегию τ - X_2. Тогда, матрицы переходных вероятностей и доходов будут следующими:

$$P = \begin{pmatrix} 0.8 & 0.2 \\ 0.4 & 0.6 \end{pmatrix} R = \begin{pmatrix} 4.5 & 2 \\ 13 & 3 \end{pmatrix}$$

Учитывая, что $F_\tau(2) = 0$, получаем систему линейных алгебраических уравнений:

$$\begin{cases} E_\tau - (1 - 0.8) * F_\tau(1) = 4 \\ E_\tau - (1 - 0.4) * F_\tau(1) = 7 \end{cases}$$

```
syms Et Ft
eqn1 = Et - (1-0.8)*Ft == 4;
eqn2 = Et - (1-0.4)*Ft == 7;
[A, B] = equationsToMatrix([eqn1, eqn2], [Et, Ft])
X = linsolve(A, B)
```

Листинг 2.3: Скрпит для решения системы уравнений

В результате выполнения скрипта matlab, было получено единственное решение:

$$E_\tau = 5/2; F_\tau(1) = -15/2$$

2.2.6 Этап(3) улучшения стратегии

Для каждого состояния S_j, где j от 1 до m, найдем допустимое решение, на котором достигается:

$$max(v_j(X_i) + \sum_{k=1}^{m} p_{jk}(X_i)F_\tau(k)$$

S_j	$\varphi_i = v_j(X_i) + p_{j1}(X_i)F_i(1)$				$max\varphi_i$	X_j
	i=1	i=2	i=3	i=4		
1	4+0.8*(-15/2)= -2	4+0.8*(-15/2)= -2	16+0.5*(-15/2)= 12.25	16+0.5*(-15/2)= 12.25	12.25	X_3, X_4
2	10.4+0.7*(-15/2)= 5.15	7+0.4*(-15/2)= 4	10.4+0.7*(-15/2)= 5.15	7+0.4*(-15/2)= 4	5.15	X_1, X_3

Так как $t \neq \tau$, то снова переходим к этапу оценивания параметров.

2.2.7 Этап(4) оценивания параметров

Выбираем стратегию τ - X_1. Тогда, матрицы переходных вероятностей и доходов будут следующими:

$$P = \begin{pmatrix} 0.8 & 0.2 \\ 0.7 & 0.3 \end{pmatrix} R = \begin{pmatrix} 4.5 & 2 \\ 11 & 9 \end{pmatrix}$$

Учитывая, что $F_\tau(2) = 0$, получаем систему линейных алгебраических уравнений:

$$\begin{cases} E_\tau - (1 - 0.8) * F_\tau(1) = 4 \\ E_\tau - (1 - 0.7) * F_\tau(1) = 10.4 \end{cases}$$

```
syms Et Ft
eqn1 = Et - (1-0.8)*Ft == 4;
eqn2 = Et - (1-0.7)*Ft == 10.4;
[A, B] = equationsToMatrix([eqn1, eqn2], [Et, Ft])
X = linsolve(A, B)
```

Листинг 2.4: Скрпит для решения системы уравнений

В результате выполнения скрипта matlab, было получено единственное решение:

$$E_\tau = -44/5; F_\tau(1) = -64$$

2.2.8 Этап(4) улучшения стратегии

Для каждого состояния S_j, где j от 1 до m, найдем допустимое решение, на котором достигается:

$$max(v_j(X_i) + \sum_{k=1}^{m} p_{jk}(X_i)F_\tau(k)$$

S_j	$\varphi_i = v_j(X_i) + p_{j1}(X_i)F_i(1)$				$max\varphi_i$	X_j
	i=1	i=2	i=3	i=4		
1	4+0.8*(-64)= -47.2	4+0.8*(-64)= -47.2	16+0.5*(-64)= -16	16+0.5*(-64)= -16	-16	X_3, X_4
2	10.4+0.7*(-64)= -34.4	7+0.4*(-64)= -18.6	10.4+0.7*(-64)= -34.4	7+0.4*(-64)= -18.6	-18.6	X_2, X_4

Итак в этапе(2) и данном, были найдены оптимальные стратегии. То есть

$$\tau = ((X_3 \text{ или } X_4), (X_2 \text{ или } X_4))^T$$

В каждом из двух состояний, имеется два варианта дальнейших действий.

Если подвести итоги, то данное решение означает что:

- В состоянии хорошего сбыта(S_1) - ничего не делать(D_3);
- В состоянии среднего сбыта(S_2) - ничего не делать(D_3).

И судя по данным итогам, ничего не делать(D_3) является лучшей стратегией. Подобный исход можно объяснить тем, что при попытках увеличения сбыта, компания тратит на это деньги и соответственно доход снижается.

Глава 3

Решение задачи анализа потокового графа с использованием методики GERT и алгебры потоковых графов

3.1 Постановка задачи

Вариант: 36.

Дано:

1. Граф GERT-сети

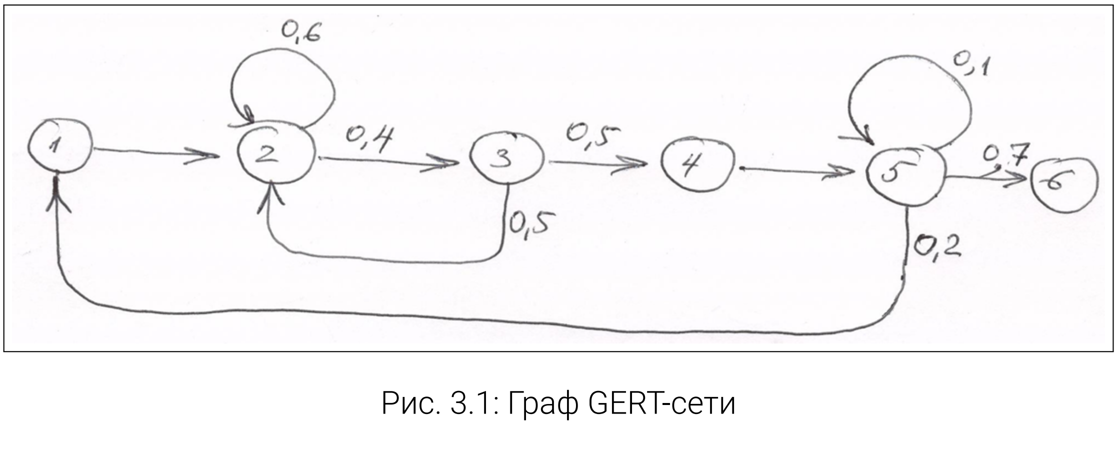

Рис. 3.1: Граф GERT-сети

2. Каждой дуге-работе (ij) поставлены в соответствие следующие данные:

   2.1. Закон распределения времени выполнения работы. Будем считать его нормальным;

2.2. Параметры закона распределения (математическое ожидание **M** и дисперсия **D**);

2.3. Вероятность p_{ij} выполнения работы, показанная на графе.

3.1.1 Задание

Часть 1

Используя методику GERT, изложенную в книге «Методы анализа сетей»
Найти:

1. Вероятность выхода в завершающий узел графа (для всех вариантов узел 6);
2. Производящую функцию длительности процесса от начального узла до завершающего узла;
3. Математическое ожидание длительности процесса от начального узла до завершающего узла;
4. Дисперсию ожидание длительности процесса от начального узла до завершающего узла;

В отчете перечислить все петли всех порядков, обнаруженные на графе, выписать уравнение Мейсона, получить решение для $W_E(S)$ и найти требуемые параметры. Примерно так, как это сделано в примере на стр. 403–409 книги Филипса и Гарсиа «Методы анализа сетей»

Часть 2

Повторить пункты задания 2, 3, 4 используя методику анализа потокового графа, основанную на обработке матрицы передач (Branch Transmittance Matrix).

Для выполнения задания рекомендуется пользоваться следующими источниками:

1. Филипс и Гарсиа «Методы анализа сетей»
2. Презентация GERT_&_Flowgraph_Algebra.pdf (выложена в ИНТРАНЕТ)
3. Ren_The Methodology of Flowgraph.pdf

3.2 Решение

3.2.1 Часть 1

Чтобы определить эквивалентную W-функцию для анализируемой GERT-сети, необходимо замкнуть сеть дугой, исходящей из узла 6 в узел 1 (рис. 3.2).

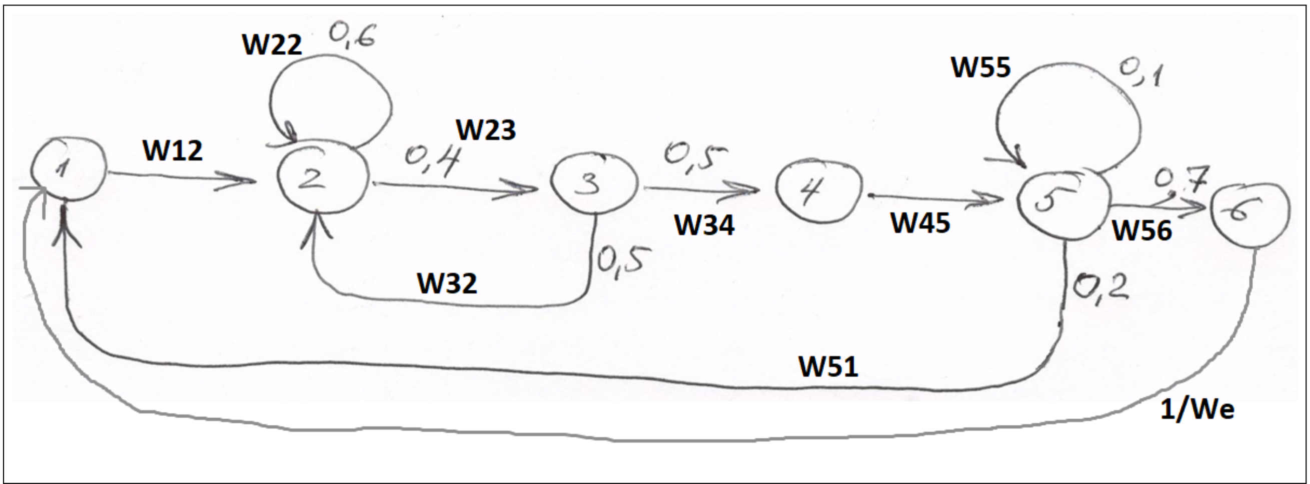

Рис. 3.2: Замкнутая GERT-сеть

Далее, выпишем в таблицу данные графа(мат. ожидание, дисперсия, W-функции)

Начало	**Конец**	**Вес ребра**	**Мат. ожидание**	**Дисперсия**	**W-функция**
1	2	1	20	9	$exp(20s + 4.5s^2)$
2	2	0.6	30	16	$0.6 * exp(30s + 8s^2)$
2	3	0.4	40	25	$0.4 * exp(40s + 12.5s^2)$
3	2	0.5	28	16	$0.5 * exp(28s + 8s^2)$
3	4	0.5	37	16	$0.5 * exp(37s + 8s^2)$
4	5	1	30	25	$exp(30s + 12.5s^2)$
5	1	0.2	30	16	$0.2 * exp(30s + 8s^2)$
5	5	0.1	10	4	$0.1 * exp(10s + 2s^2)$
5	6	0.7	30	16	$0.7 * exp(30s + 8s^2)$

Таблица 3.1: Данные анализируемой GERT-сети

Петли первого порядка:

- $W_{12}W_{23}W_{34}W_{45}W_{51}$;

- $W_{12}W_{23}W_{34}W_{45}W_{56}\frac{1}{W_e}$;
- W_{22};
- $W_{23}W_{32}$;
- W_{55};

Петли второго порядка:

- $W_{22}W_{55}$;
- $W_{55}W_{23}W_{32}$;

Уравнение Мейсона:

$$H = 1 - W_{12}W_{23}W_{34}W_{45}W_{51} - W_{12}W_{23}W_{34}W_{45}W_{56}\frac{1}{W_e} - W_{22} - W_{23}W_{32} - W_{55} + W_{22}W_{55} + W_{55}W_{23}W_{32} = 0$$

Выведем $W_E(S)$:

$$1-W_{12}W_{23}W_{34}W_{45}W_{51}-W_{22}-W_{23}W_{32}-W_{55}+W_{22}W_{55}+W_{55}W_{23}W_{32} = W_{12}W_{23}W_{34}W_{45}W_{56}\frac{1}{W_e}$$

$$W_E(S) = (W_{12}W_{23}W_{34}W_{45}W_{56})/(1-W_{12}W_{23}W_{34}W_{45}W_{51}-W_{22}-W_{23}W_{32}-W_{55}+W_{22}W_{55}+W_{55}W_{23}W_{32})$$

Вычислим математическое ожидание и дисперсию: $M_E(s) = 1$ при $s = 0$

Так как $W_E(s) = p_E M_E(s)$, то $p_E = W_E(0)$, тогда $M_E(s) = \frac{W_E(s)}{p_E} = \frac{W_E(s)}{W_E(0)}$

Вычисляя первую и вторую производные по s функции $M_E(s)$, и полагая $s = 0$, находим математическое ожидание:

$$\mu_{1E} = \frac{\partial M_E(s)}{\partial s}|s = 0$$

и дисперсию:

$$\sigma^2 = \mu_{2E} - [\mu_{1E}]^2$$

Вероятность выхода в завершающий узел графа:

$$p_E = W_E(0)$$

Для решения задачи был написан скрипт matlab, код приведен в приложении 4.

```
We =
-(7*exp((s*(91*s + 314))/2))/(5*exp(2*s*(s + 5)) - 3*exp(10*s*(s +
   ↪  4)) + 30*exp(2*s*(4*s + 15)) - exp((3*s*(15*s + 52))/2) +
   ↪ 10*exp((s*(41*s + 136))/2) + 2*exp((s*(91*s + 314))/2) - 50)

We0 =
1

m1 =
2845/7

m2 =
11938987/49

D =
3844962/49
```

Листинг 3.1: Результат

Были получены следующие результаты:

1. Вероятность выхода в завершающий узел графа равна 100% ($p = W_E = 1$).
2. Математическое ожидание 406,43.
3. Дисперсия времени выхода процесса в завершающий узел графа 78 468,61.

3.2.2 Часть 2

Алгоритм дальнейших действий основан на:

- Презентация GERT_&_Flowgraph_Algebra.pdf (со слайда 56);
- Ren_The Methodology of Flowgraph.pdf (со страницы 35).

Определим матрицу Q, не забывая про обратную связь.

$$Q = \begin{pmatrix} 0 & q_{12} & 0 & 0 & 0 & 0 \\ 0 & q_{22} & q_{23} & 0 & 0 & 0 \\ 0 & q_{32} & 0 & q_{34} & 0 & 0 \\ 0 & 0 & 0 & 0 & q_{45} & 0 \\ q_{51} & 0 & 0 & 0 & q_{55} & q_{56} \\ w_{61} & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

Определим матрицу коэффициентов $A = I_6 - Q^T$.

$$A = \begin{pmatrix} 1 & 0 & 0 & 0 & -q_{51} & -w_{61} \\ -q_{12} & 1-q_{22} & -q_{32} & 0 & 0 & 0 \\ 0 & -q_{23} & 1 & 0 & 0 & 0 \\ 0 & 0 & -q_{34} & 1 & 0 & 0 \\ 0 & 0 & 0 & -q_{45} & 1-q_{55} & 0 \\ 0 & 0 & 0 & 0 & -q_{56} & 1 \end{pmatrix}$$

Находим

$$det(A)$$

далее

$$\frac{\partial det(A)}{\partial w_{61}}$$

$$det(A|w_{61}=0)$$

Далее можно вывести $W_E(S)$ с помощью формулы:

$$W_E(S) = -\frac{\frac{\partial det(A)}{\partial w_{61}}}{det(A|w_{61}=0)}$$

Для расчетов, был написан matlab скрипт, код представлен в приложении 5.

```
[     1,        0,     0,     0,     -q51, -w61]
[ -q12, 1 - q22, -q32,     0,        0,     0]
[     0,    -q23,     1,     0,        0,     0]
[     0,        0, -q34,     1,        0,     0]
[     0,        0,     0, -q45, 1 - q55,     0]
[     0,        0,     0,     0,     -q56,     1]



-(q12*q23*q34*q45*q56)/(q22 + q55 + q23*q32 - q22*q55 - q23*q32*
    ↪ q55 + q12*q23*q34*q45*q51 - 1)
```

Листинг 3.2: Результат

Во второй строчке был получен $W_E(S)$, который полностью(за исключением знаков) совпадает с $W_E(S)$ найденным в части 1.

Далее, имея $W_E(S)$ находим необходимые переменные.

Для расчетов, был использован скрипт из приложения 6.

```
We =
-(7*exp((s*(91*s + 314))/2))/(5*exp(2*s*(s + 5)) - 3*exp(10*s*(s +
    ↪  4)) + 30*exp(2*s*(4*s + 15)) - exp((3*s*(15*s + 52))/2) +
    ↪ 10*exp((s*(41*s + 136))/2) + 2*exp((s*(91*s + 314))/2) - 50)

We0 =
1

m1 =
2845/7

m2 =
11938987/49

D =
3844962/49
```

Листинг 3.3: Результат

Были получены следующие результаты:

1. Вероятность выхода в завершающий узел графа равна 100% ($p = W_E = 1$).
2. Математическое ожидание 406,43.
3. Дисперсия времени выхода процесса в завершающий узел графа 78 468,61.

Которые полностью совпадает с результатами части 1.

Глава 4

Поиск оптимальных параметров сети систем массового обслуживания

4.1 Постановка задачи

Вариант: задача 4, вариант 144.

№ вар	$Q = \{q_{ij}\}_{\substack{i=\overline{0,n}\\ j=\overline{0,n}}}$	ca_0	λ_0	L_r	μ	$\{cs_j\}$
144	$\begin{array}{c\|c\|c\|c\|c} 0 & 0.2 & 0.3 & 0.2 & 0.3 \\ \hline 0.1 & 0 & 0.2 & 0.6 & 0.1 \\ \hline 0.6 & 0.2 & 0 & 0.1 & 0.1 \\ \hline 0 & 0.5 & 0.1 & 0 & 0.4 \\ \hline 0.5 & 0.3 & 0.1 & 0.1 & 0 \end{array}$	0.16	8	-	10	$0.04 \mid 0.04 \mid 0.04 \mid 0.04$

Найти:

$$minL(\mu) = \sum_{j=1}^{n} L_j$$

При условии:

$$\sum_{j=1}^{n} \mu_j = \mu$$

4.2 Решение

Вычислим мощность $\mu > \mu_j^1, cs$ и ca для каждой станции.

Скорость прихода задач в узел j: $\lambda_j = \lambda_{0j} + \sum_{i=0}^{n} q_{ij}\lambda_i, j = 0, ..., n$

$$Q = \begin{pmatrix} 0 & 0.2 & 0.3 & 0.2 & 0.3 \\ 0.1 & 0 & 0.2 & 0.6 & 0.1 \\ 0.6 & 0.2 & 0 & 0.1 & 0.1 \\ 0 & 0.5 & 0.1 & 0 & 0.4 \\ 0.5 & 0.3 & 0.1 & 0.1 & 0 \end{pmatrix}$$

$\lambda_0 = 8$

$\lambda_1 = 0.2\lambda_0 + 0\lambda_1 + 0.2\lambda_2 + 0.5\lambda_3 + 0.3\lambda_4$

$\lambda_2 = 0.3\lambda_0 + 0.2\lambda_1 + 0\lambda_2 + 0.1\lambda_3 + 0.1\lambda_4$

$\lambda_3 = 0.2\lambda_0 + 0.6\lambda_1 + 0.1\lambda_2 + 0\lambda_3 + 0.1\lambda_4$

$\lambda_4 = 0.3\lambda_0 + 0.1\lambda_1 + 0.1\lambda_2 + 0.4\lambda_3 + 0\lambda_4$

```
A = [1 0 0 0 0;
 0.2 -1 0.2 0.5 0.3;
 0.3 0.2 -1 0.1 0.1;
 0.2 0.6 0.1 -1 0.1;
 0.3 0.1 0.1 0.4 -1];
b=[8; 0; 0; 0; 0];
lambdaj=A\b
```

Листинг 4.1: Код Matlab

```
lambdaj =
    8.0000
    9.1128
    5.7833
    8.3697
    7.2375
```

Листинг 4.2: Результат

Проверим полученный результат:

```
>> [ 0 0.1 0.6 0 0.5] * lambdaj


ans =
    8.0000
```

Листинг 4.3: Проверка

Как и ожидалось, была получена λ_0.

Вычислим ca_j. Для этого, сперва найдем все λ_{ij}, где $\lambda_{ij} = \lambda_i * q_{ij}$.

```
N=5;
Q = [0 0.2 0.3 0.2 0.3;
 0.1 0 0.2 0.6 0.1;
 0.6 0.2 0 0.1 0.1;
 0 0.5 0.1 0 0.4;
 0.5 0.3 0.1 0.1 0];
lambdaij=[];
for i = 1:N
 for j = 1:N
 lambdaij(i,j) = lambdaj(i)*Q(i, j);
 end
end
 lambdaij
```

Листинг 4.4: Код Matlab

```
lambdaij =


         0    1.6000    2.4000    1.6000    2.4000
    0.9113         0    1.8226    5.4677    0.9113
    3.4700    1.1567         0    0.5783    0.5783
         0    4.1849    0.8370         0    3.3479
    3.6188    2.1713    0.7238    0.7238         0
```

Листинг 4.5: Результат

$$\lambda = \begin{pmatrix} 0 & 1.6 & 2.4 & 1.6 & 2.4 \\ 0.91 & 0 & 1.82 & 5.47 & 0.91 \\ 3.47 & 1.16 & 0 & 0.58 & 0.58 \\ 0 & 4.18 & 0.84 & 0 & 3.35 \\ 3.62 & 2.17 & 0.72 & 0.72 & 0 \end{pmatrix}$$

Решим уравнения по формулам:

$$ca_j = \frac{\lambda_{0j}}{\lambda_j} ca_{0j} + \sum_{i=1}^{n} \frac{\lambda_{ij}}{\lambda_j} ca_{ij} = \sum_{i=0}^{n} \frac{\lambda_{ij}}{\lambda_j} ca_{ij}$$

$$cd_{ij} = q_{ij} cd_{ij} + 1 - q_{ij}$$

```
caA = 0;
 caB = [0 0 0 0 0];
for j = 1:N
 for i = 1:N
 caA(j,i) = lambdaij(i,j)/lambdaj(j)*Q(i, j);
 caB(j)=caB(j)+lambdaij(i,j)/lambdaj(j)*(1-Q(i,j));
 end
end
 caA = caA-eye(5);
 caA(1,:) = [1 0 0 0 0];
 caA;
 caA^-1;
 caB = -caB';
 caB(1)=0.49;
 caj = (caA^-1)*caB
```

Листинг 4.6: Код Matlab

```
caj =
    0.1600
    0.9486
    0.8902
    0.9461
    0.9049
```

Листинг 4.7: Результат

Вычислим L_j и P_j

$$p_j = \frac{\lambda_j}{\mu_j m_j}$$

$$L_j(\lambda_j, ca_j, \mu_j, cs_j) = \frac{(\frac{\lambda_j}{\mu_j})^2 (ca_j + cs_j) * g(\lambda_j, ca_j, \mu_j, cs_j)}{2(1 - \frac{\lambda_j}{\mu_j})} + \frac{\lambda_j}{\mu_j}$$

$$PI_j(\mu_j) = -V_j \frac{\partial L_j(\mu_j)}{\partial(\mu_j)}$$

Где:

$$\lambda_j = 9.1128, ca_0 = 0.16, cs_1 = 0.04$$

Для этого был написан скрипт matlab.

```
for i = 2:N
 [Lj(i-1), Pj(i-1)] = params(lambdaj(i), caj(i), m(i-1), cs(i-1));
end
 Lj
 Pj
 L = sum(Lj)

function [ fLj, fPj ] = params( fl, fca, fm, fcs )
 Lj = '(l/m)^2*(ca+csj)*exp(-2*(1-l/m)*(1-ca)^2/(3*(l/m)*(ca+csj))
   ↪ )/(2*(1-l/m))';
 syms m;
 syms l;
 syms ca;
 syms csj;
 fLj = subs(Lj,l, fl);
 fLj = subs(fLj,m, fm);
 fLj = subs(fLj,ca, fca);
 fLj = subs(fLj,csj, fcs);
 fLj = vpa(fLj);

 Pj = '-((l)/(l-m)^2)';
 fPj = subs(Pj,l, fl);
 fPj = subs(fPj,m, fm);
 fPj = -1*vpa(fPj);
 end
```

Листинг 4.8: Код Matlab

```
Lj =
[ 4.6255839976370720273840674915939,
   ↪ 0.36659806895060537891974885952995,
   ↪ 2.1179331569743248761799734854413,
   ↪ 0.89370855816402942385738205100272]



Pj =
[ 11.576845733984487216317150244183,
   ↪ 0.32525589689516798523377343071108,
```

```
    ↪ 3.1492188330322032096928472738867,
    ↪ 0.94838805064508015837789082867662]

L =
8.0038237817260317063411718875679
```
Листинг 4.9: Результат

Воспользуемся следующей формулой:

$$PI_j(\mu_j, (\lambda_j + \varepsilon_j)) = max\{PI_j(\mu_j), j \in J_0\}$$

Чем выше загрузка узла J, тем больше $PI_j = -\frac{\partial L(\mu_j)}{\partial \mu_j}$
Для узла **M/M/1** имеем $PI_j = -\frac{\partial L(\mu_j)}{\partial \mu_j} = \frac{\lambda_j}{(\lambda_j - \mu_j)^2}$. Благодаря расчётам по этой формуле, можно понять в каких узла нужно увеличивать или уменьшать интенсивность входного потока.
Далее, применим следующий алгоритм:

Найти:

$$j_2 = \arg\max_{(j)} PI_j(\mu_j) \;;$$

$$j_1 = \arg\min_{(j)} PI_j(\mu_j) \;;$$

если:

$j_1 \in J_1$, **выполнить** $J_0 \leftarrow J_0 - \{j_1\}$;

$j_2 \in J_2$, **выполнить** $J_0 \leftarrow J_0 - \{j_2\}$;

$j_1 \notin J_1$ **и** $j_2 \notin J_2$, **найти** $\Delta_1 = \min\{\Delta,\ \mu_{j_1} - \lambda_{j_1} - \varepsilon_{j_1}\}$ **и**

выполнить $\mu_{j_2} \leftarrow \mu_{j_2} + \Delta_1$, $J_1 \leftarrow J_1 \cup \{j_2\}$ **и** $J_2 \leftarrow J2 \cup \{j_1\}$,

идти к шагу 4.

Если J_0 **есть пустое множество или** $PI_{j_1} = PI_{j_2}$,

Как значение Δ возьмем 0.5.
По приведенному выше алгоритму определяем множества J_0, J_1, J_2. Результаты приведены в таблице 4.2.

№	J_0	J_1	J_2	Действия
1	1,2,3,4	-	-	$J_1 \leftarrow 1$ $J_2 \leftarrow 2$
2	1,2,3,4	1	2	$J_1 \leftarrow 1$ $J_2 \leftarrow 2$
3	1,2,3,4	1	2	$J_1 \leftarrow 3$ $J_2 \leftarrow 2$
4	1,2,3,4	1,3	2	$J_1 \leftarrow 1$ $J_2 \leftarrow 2$
5	1,2,3,4	1,3	2	$J_1 \leftarrow 3$ $J_2 \leftarrow 4$
6	1,2,3,4	1,3	2,4	$J_1 \leftarrow 1$ $J_2 \leftarrow 2$
7	1,2,3,4	1,3	2,4	$J_0 \leftarrow J_0 - 1$ $J_0 \leftarrow J_0 - 2$ $J_1 \leftarrow 2$ $J_2 \leftarrow 1$
8	3,4	1,2,3	1,2,4	$J_1 \leftarrow 4$ $J_2 \leftarrow 3$
9	3,4	1,2,3,4	1,2,3,4	$J_0 \leftarrow J_0 - 3$ $J_0 \leftarrow J_0 - 4$ $J_1 \leftarrow 3$ $J_2 \leftarrow 4$
10	-	1,2,3,4	1,2,3,4	-

Таблица 4.2: Формирование множеств J

Результаты перераспределения мощностей представлено в таблице 4.3.

№	μ_1	μ_2	μ_3	μ_4	Действия
1	10	10	10	10	$\mu_2 - \Delta$ $\mu_1 + \Delta$
2	10.5	9.5	10	10	$\mu_2 - \Delta$ $\mu_1 + \Delta$

3	11	9	10	10	$\mu_2 - \Delta$ $\mu_3 + \Delta$
4	11	8.5	10.5	10	$\mu_2 - \Delta$ $\mu_1 + \Delta$
5	11.5	8	10.5	10	$\mu_4 - \Delta$ $\mu_3 + \Delta$
6	11.5	8	11	9.5	$\mu_2 - \Delta$ $\mu_1 + \Delta$
7	12	7.5	11	9.5	$\mu_1 - \Delta$ $\mu_2 + \Delta$
8	11.5	8	11	9.5	$\mu_3 - \Delta$ $\mu_4 + \Delta$
9	11.5	8	10.5	10	$\mu_4 - \Delta$ $\mu_3 + \Delta$
10	11.5	8	11	9.5	-

Таблица 4.3: Перераспределение мощностей

Пересчитаем значения PI_j, L_j. Результаты вычисления PI_j приведены в таблице 4.4 и 4.5.

№	PI_1	PI_2	PI_3	PI_4
1	11.5768	0.3252	3.1492	0.9483
2	4.7354	0.4186	3.1492	0.9483
3	2.5586	0.5589	3.1492	0.9483
4	2.5586	0.7835	1.8443	0.9483
5	1.5990	1.1769	1.8443	0.9483
6	1.5990	1.1769	1.2098	1.4138
7	1.0931	1.9623	1.2098	1.4138
8	1.5990	1.1769	1.2098	1.4138
9	1.5990	1.1769	1.8443	0.9483
10	1.5990	1.1769	1.2098	1.4138

Таблица 4.4: Пересчитанное PI_j

№	L_1	L_2	L_3	L_4	L
1	4.6255	0.3665	2.1179	0.8937	8.0038
2	2.8172	0.4381	2.1179	0.8937	6.2669
3	1.9765	0.5347	2.1179	0.8937	5.5229
4	1.9765	0.6709	1.5434	0.8937	5.0845
5	1.4944	0.8743	1.5434	0.8937	4.8059
6	1.4944	0.8743	1.1930	1.1491	4.7109
7	1.1840	1.2051	1.1930	1.1491	4.7313
8	1.4944	0.8743	1.1930	1.1491	4.7109
9	1.4944	0.8743	1.5434	0.8937	4.8059
10	1.4944	0.8743	1.1930	1.1491	4.7109

Таблица 4.5: Пересчитанное L_j и L

4.3 Вывод

В данной работе была произведена оптимизация ССМО, в частности перераспределение мощностей в системе, для уменьшения очереди.

Начальные значения:
$\mu = (10, 10, 10, 10)$
$L = 8.0038$
$L_j = (4.6255, 0.3655, 2.1179, 0.8937)$

После оптимизации:
$\mu = (10.5, 8, 11, 9.5)$
$L = 4.7109$
$L_j = (1.4944, 0.8743, 1.1930, 1.1491)$

Как и ожидалась, после оптимизации сети, загруженность системы понизилась.

В данной части работы была произведена оптимизация ССМО, в частности перераспределение мощностей в системе, для уменьшения очереди.

Начальные значения:
$\mu = (10, 10, 10, 10)$

$L = 8.0038$
$L_j = (4.6255, 0.3655, 2.1179, 0.8937)$

После оптимизации:
$\mu = (10.5, 8, 11, 9.5)$
$L = 4.7109$
$L_j = (1.4944, 0.8743, 1.1930, 1.1491)$

Как и ожидалась, после оптимизации сети, загруженность системы понизилась.

Заключение

В работе были рассмотрены различные математические модели для решения задачи выбора оптимального решения.

При решении задачи стохастического программирования, было выяснено, что увеличение доверительной вероятности α приводит к ухудшению оценок получаемых решений по первому и второму критерию. При $\alpha = 0.5, K_\alpha = 0$в случае чего получается решение, совпадающее с решением задачи с детерминированными ограничениями.

При $\alpha < 0.5$ квантиль приобретает отрицательное значение. За счет чего ограничения ослабевают(их выполнение становится менее важным), а при $\alpha > 0.5$ ограничения становяться более жесткими.

При решении задачи анализа потокового графа, было применено два подходам: с использованием методики GERT и с использованием алгебры потоковых графов. По итогу, оба метода выдали верный результат(прохождение графа). Однако, использование алгебры потоковых графов, на мой взляд, является более простым методом, так как необходимости в поиске петель различных порядков, а также в работе с W-функциями.

В задаче поиска оптимальных параметров сети системы массового обслуживания, была произведена оптимизация ССМО, путем перераспределение мощностей в системе, для уменьшения очереди. Что в итоге привело к понижению загруженности системы.

Список использованных источников

1. Д. Филлипс, А. Гарсиа-Диас. Методы анализа сетей: Пер. с англ. — М.: Мир, 1984.— 496 с, ил.
2. Ren, Yu. The methodology of flowgraph models. PhD thesis, The London School of Economics and Political Science (LSE). –– 2011.
3. Сиднев А. Г. Системный анализ. Часть 1 [Электронный ресурс] // Интранет- портал ИИТУ СПбПУ. 2018. URL: http://intranet.ftk.spbstu.ru/docinfo.php?InfoFtkDocumentID=1386947 (дата обращения: 22.04.2018)
4. Горбунов В.М. Теория принятия решений: учебное пособие. Томск: Изд-во Томск. политех. ун-та, 2010.
5. Системный анализ и принятие решений: учебное пособие / Е.Н. Бендерская, Д.Н. Колесников, В.И. Пахомова и др.; Под ред. Д.Н. Колесникова. 2-е изд. СПб.: Изд-во СПбГПУ, 2001.
6. Optimization Toolbox: описание функции FGOALATTAIN [Электронный ре- сурс] // MATLAB.Exponenta, центр компетенций MathWorks. URL: http://matlab.exponenta.ru/optimiz/book_4/2/fgoalattain.php (дата обращения: 17.02.2018).
7. Таха Х. А. Введение в исследование операций: Пер. с англ. 7-е изд. М.: Изда-тельский дом «Вильямс», 2005.

Приложение 1

Matlab скрипт с описанием задачи

```
clc; clearvars
% Параметры
Tmin1 = (18*60)*0.7;  %Минимально допустимое время работы 1 станка
Tmin2 = (20*60)*0.7;  %Минимально допустимое время работы 2 станка
Tmax1 = (18*60);  %Максимально допустимое время работы 1 станка
Tmax2 = (20*60);  %Максимально допустимое время работы 2 станка

% Целевые функции
f1 = @(X) 30*X(1) + 20*X(3); % -> max
f2 = @(X) 3*(X(1)+X(2))+12*(X(3)+X(4)); % -> min
f3 = @(X) 45*X(2) + 21*X(4); % -> max

z1 = @(N) -f1(N); % -> min
z3 = @(N) -f3(N); % -> min

% Функциональные ограничения в( данном случае только линейные)
A = [1,1,5,5;
     -1,-1,-5,-5;
     2,2,7,7;
     -2,-2,-7,-7;
     1,1,0,0;
     0,0,1,1];
b = [Tmax1; -Tmin1; Tmax2; -Tmin2;  5000; 9000];

Aeq = [];
beq = [];
% Параметрические ограничения
lb = [0; 0; 0; 0];
ub = [5000; 5000; 9000; 9000];
```

Листинг 4.10: Скрипт с описанием задачи

Приложение 2

Matlab скрипт выделения главного критерия

```
clc; clearvars
% Параметры
Tmin1 = (18*60)*0.7;  %Минимально допустимое время работы 1 станка
```

```
Tmin2 = (20*60)*0.7;  %Минимально допустимое время работы 2 станка
Tmax1 = (18*60);  %Максимально допустимое время работы 1 станка
Tmax2 = (20*60);  %Максимально допустимое время работы 2 станка
coeff = 0.7;

% Целевые функции
f1 = @(X) 30*X(1) + 20*X(3); % -> max
f2 = @(X) 3*(X(1)+X(2))+12*(X(3)+X(4)); % -> min
f3 = @(X) 45*X(2) + 21*X(4); % -> max

z1 = @(N) -f1(N); % -> min
z3 = @(N) -f3(N); % -> min

% Функциональные ограничения в( данном случае только линейные)
A = [1,1,5,5;
     -1,-1,-5,-5;
     2,2,7,7;
     -2,-2,-7,-7;
     1,1,0,0;
     0,0,1,1
     3,3,12,12;
     0,-45,0,-21];
b = [Tmax1; -Tmin1; Tmax2; -Tmin2;  5000; 9000; 1950; -1000];

Aeq = [];
beq = [];
% Параметрические ограничения
lb = [0; 0; 0; 0];
ub = [5000; 5000; 9000; 9000];

%% Поиск оптимумов частных критериев
startingPoint = lb;
[x, z1_opt] = fmincon(z1, startingPoint, A, b, Aeq, beq, lb, ub)
[x, f2_opt] = fmincon(f2, startingPoint, A, b, Aeq, beq, lb, ub)
[x, z3_opt] = fmincon(z3, startingPoint, A, b, Aeq, beq, lb, ub)


[x, f_val] = fmincon(z1, startingPoint, A, b, Aeq, beq, lb, ub)
```
Листинг 4.11: Скрипт с выделением главного критерия

Приложение 3

Решение задачи стохастического программирования

```
clc; clearvars
% Параметры
Tmin1 = (18*60)*0.7;  %Минимально допустимое время работы 1 станка
Tmin2 = (20*60)*0.7;  %Минимально допустимое время работы 2 станка
Tmax1 = (18*60);  %Максимально допустимое время работы 1 станка
Tmax2 = (20*60);  %Максимально допустимое время работы 2 станка

% Целевые функции
f1 = @(X) 30*X(1) + 20*X(3); % -> max
f2 = @(X) 3*(X(1)+X(2))+12*(X(3)+X(4)); % -> min
f3 = @(X) 45*X(2) + 21*X(4); % -> max
z1 = @(N) -f1(N); % -> min
z3 = @(N) -f3(N); % -> min

% Обобщенный критерий
z1_opt = -9160;
f2_opt = 1814.4;
z3_opt = -12804;

f = @(N) (1-z1(N)/z1_opt)^2+(1-f2(N)/f2_opt)^2+(1-z3(N)/z3_opt)^2;
% Функциональные ограничения
A = [
    1,1,5,5;
    -1,-1,-5,-5;
    -2,-2,-7,-7;
    1,1,0,0;
    0,0,1,1];
b = [Tmax1;-Tmin1;  -Tmin2;  5000; 9000];
Aeq = [];
beq = [];
% Параметрические ограничения
lb = [0; 0; 0; 0];
ub = [5000; 5000; 9000; 9000];
% Опции оптимизатора
options = optimoptions('fmincon','Display','none');
startingPoint = [0, 0, 0, 0];
% Параметры для задачи стохастического программирования
alpha = 0.1:0.1:0.9;
Ka = icdf('Normal', alpha, 0, 1); % Квантили распределения N(0,1)
global K
global m1; global m2; global m3; global m4
```

```
global d1; global d2; global d3; global d4
m1=2;m2=2;m3=7;m4=7;
d1=1;d2=1;d3=3.5;d4=3.5;
% Решение детерминированной задачи:
[Ndet, ~] = fmincon(f, startingPoint,[2,2,7,7; A], [Tmax2; b], Aeq, beq, lb, ub
    ↪ , [], options);
% Решение стохастической задачи:
N = zeros(4,numel(alpha));
for i = 1:numel(alpha)
 K = Ka(i);
 [N(:,i), ~] = fmincon(f, startingPoint, ...
 A, b, Aeq, beq, lb, ub, @nonlin, options);
end
%% Вывод результатов
F1 = cellfun(f1, num2cell(N,1));
F2 = cellfun(f2, num2cell(N,1));
F3 = cellfun(f3, num2cell(N,1));
fprintf('alpha = %.1f, X_11 = %6.4f, X_12 = %6.4f, x_21 = %6.4f, x_22 = %6.3f,
    ↪ f1 = %.2f, f2 = %.0f, f3 = %.2f\n', ...
 [alpha; N(1,:); N(2,:); N(3,:); N(4,:); F1; F2; F3])
fprintf('Для детерминированных ограничений:\n')
fprintf('X_11 = %3.4f, X_12 = %3.4f, x_21 = %6.4f, x_22 = %6.3f, f1 = %.2f, f2
    ↪ = %.0f, f3 = %.2f\n', ...
 Ndet(1), Ndet(2), Ndet(3), Ndet(4), f1(Ndet), f2(Ndet), f3(Ndet))
%% Функция, задающая нелинейные ограничения
function [c, ceq] = nonlin(N)
global K;
global m1; global m2; global m3; global m4
global d1; global d2; global d3; global d4
  c = m1*N(1)+m2*N(2) + m3*N(3)+m4*N(4) + K*sqrt(d1*N(1)^2 + d2*N(2)^2 + d3*N
    ↪ (3)^2 + d4*N(4)^2) - 1200;
 ceq = [];
end
```

Листинг 4.12: Решение задачи стохастического программирования

Приложение 4

Скрипт для нахождения мат. ожидания, производных, дисперсии

```
clc
%Исходные данные
%M — математическое ожидание
```

```
%D — дисперсия
%P — вероятность
P12 = 1; M12 = 20; D12 = 9;
P22 = 0.6; M22 = 30; D22 = 16;
P23 = 0.4; M23 = 40; D23 = 25;
P32 = 0.5; M32 = 28; D32 = 16;
P34 = 0.5; M34 = 37; D34 = 16;
P45 = 1; M45 = 30; D45 = 25;
P51 = 0.2; M51 = 30; D51 = 16;
P55 = 0.1; M55 = 10; D55 = 4;
P56 = 0.7; M56 = 30; D56 = 16;

syms s
%W функции
W12 = P12*exp(M12*s+D12/2*s^2);
W22 = P22*exp(M22*s+D22/2*s^2);
W23 = P23*exp(M23*s+D23/2*s^2);
W32 = P32*exp(M32*s+D32/2*s^2);
W34 = P34*exp(M34*s+D34/2*s^2);
W45 = P45*exp(M45*s+D45/2*s^2);
W51 = P51*exp(M51*s+D52/2*s^2);
W55 = P55*exp(M55*s+D55/2*s^2);
W56 = P56*exp(M56*s+D56/2*s^2);

We = (W12*W23*W34*W45*W56)/(1 - W12*W23*W34*W45*W51 - W22 - W23*W32 - W55+W22*
    ↪ W55+W55*W23*W32);

We = simplify(We)
We0 = subs(We, 's', 0)  % We(0)

% Нахождение мат. ожидания и дисперсии
Me = We/We0;

% Нахождение производной первого порядка при s=0
m1 = diff(Me, 's');
m1 = subs(m1, 's', 0)    % Замена символа s на 0 в выражении m1

% Нахождение производной второго порядка при s=0
m2 = diff(Me, 's',2);
m2=subs(m2, 's', 0)      % Замена символа s на 0 в выражении m2

% Нахождение дисперсии времени выхода процесса в завершающий узел графа
D = m2 - (m1)^2
```
Листинг 4.13: Код Matlab

Приложение 5

Алгебра потоковых графов

```
clc; clearvars

syms q12
syms q22
syms q23
syms q32
syms q34
syms q45
syms q51
syms q55
syms q56
syms w61
syms s

Q=[0 q12 0 0 0 0;
   0 q22 q23 0 0 0;
   0 q32 0 q34 0 0;
   0 0 0 0 q45 0;
   q51 0 0 0 q55 q56;
   w61 0 0 0 0 0];

A1 = eye(size(Q,1)) - transpose(Q);
disp(A1);

det_A1 = det(A1);

det_dw=diff(det_A1, w61);

det2_A1=subs(det_A1, w61, 0);

We= -det_dw/det2_A1;
disp(We);
```

Листинг 4.14: Matlab скрипт

Приложение 6

Скрипт для нахождения мат. ожидания, производных, дисперсии

```
clc; clearvars

%M — математическое ожидание
%D — дисперсия
%P — вероятность
P12 = 1; M12 = 20; D12 = 9;
P22 = 0.6; M22 = 30; D22 = 16;
P23 = 0.4; M23 = 40; D23 = 25;
P32 = 0.5; M32 = 28; D32 = 16;
P34 = 0.5; M34 = 37; D34 = 16;
P45 = 1; M45 = 30; D45 = 25;
P51 = 0.2; M51 = 30; D51 = 16;
P55 = 0.1; M55 = 10; D55 = 4;
P56 = 0.7; M56 = 30; D56 = 16;

syms q12
syms q22
syms q23
syms q32
syms q34
syms q45
syms q51
syms q55
syms q56
syms w61
syms s

Q=[0 q12 0 0 0 0;
   0 q22 q23 0 0 0;
   0 q32 0 q34 0 0;
   0 0 0 0 q45 0;
   q51 0 0 0 q55 q56;
   w61 0 0 0 0 0];

A1 = eye(size(Q,1)) - transpose(Q);
disp(A1);

det_A1 = det(A1);
disp(det_A1);

det_dw=diff(det_A1, w61);
```

```
disp(det_dw);

det2_A1=subs(det_A1, w61, 0);
disp(det2_A1);

We= -det_dw/det2_A1;
disp(We);


syms s

We=subs(We, q12, P12*exp(M12*s+D12/2*s^2));
We=subs(We, q22, P22*exp(M22*s+D22/2*s^2));
We=subs(We, q23, P23*exp(M23*s+D23/2*s^2));
We=subs(We, q32, P32*exp(M32*s+D32/2*s^2));
We=subs(We, q34, P34*exp(M34*s+D34/2*s^2));
We=subs(We, q45, P45*exp(M45*s+D45/2*s^2));
We=subs(We, q51, P51*exp(M51*s+D51/2*s^2));
We=subs(We, q55, P55*exp(M55*s+D55/2*s^2));
We=subs(We, q56, P56*exp(M56*s+D56/2*s^2));

We = simplify(We)
We0 = subs(We, 's', 0)  % We(0)

% Нахождение мат. ожидания и дисперсии
Me = We/We0;

% Нахождение производной го1- порядка при s=0
m1 = diff(Me, 's');
m1 = subs(m1, 's', 0)    % Замена символа s на 0 в выражении m1

% Нахождение производной го2- порядка при s=0
m2 = diff(Me, 's',2);
m2=subs(m2, 's', 0)      % Замена символа s на 0 в выражении m2

% Нахождение дисперсии времени выхода процесса в завершающий узел графа
D = m2 - (m1)^2
```
Листинг 4.15: Matlab скрипт