```
text(.5-q, .5+2*q, 'Class A');
text(.5+q, .5+2*q, 'Class B');
text(.5+q, .5-2*q, 'Class A');
text(.5-q, .5-2*q, 'Class B');
% Определяем метки для классов
a = -1, c = -1, b = 1, d = 1;
% Формируем входной вектор
P = [A B C D];
% Формируем вектор ответов
T = [repmat(a, 1, length(A)) repmat(b, 1, length(B))  repmat(c, 1, length(C)) repmat(d, 1, length(D))];
% Определяем сеть прямого распространения
net = feedforwardnet([5 3]);
% Вся выборка под обучающие примеры
net.divideParam.trainRatio = 1;
% Нет валидационной выборки
net.divideParam.valRatio = 0;
% Нет тестовой выборки
net.divideParam.testRatio = 0;
% Обучаем сеть
[net, tr, Y, E] = train(net, P, T);
view(net);
figure(2);
% График ожидаемых ответов сети
% график получается зубчатым, т.к. метки для классов
%определены как -1 и +1
plot(T', 'linewidth', 2);
hold on;
% График реальных ответов сети
plot(Y', 'r--');
grid on;
% Рисуем легенду к графикам
legend('Targets', 'Network response', 'location', 'best');
% Определяем видимый диапазон по оси y
ylim([-1.25 1.25]);
% Создаём  набор 601 числа
span = -1:.005:2;
% P1 и P2 станут матрицами размером 601x601
[P1, P2] = meshgrid(span, span);
% Транспонируем вектор размером 601*601 = 361201
pp = [P1(:) P2(:)]';
% Получаем выходы сети
aa = net(pp);
% Возвращаемся к первой канве
figure(1);
% Рисуем трёхмерную сетку
mesh(P1, P2, reshape(aa, length(span), length(span))-5);
% Подбираем лучшее сочетание цветов
colormap cool;
```


На рисунке 5.22 показаны паттерны входных классов.

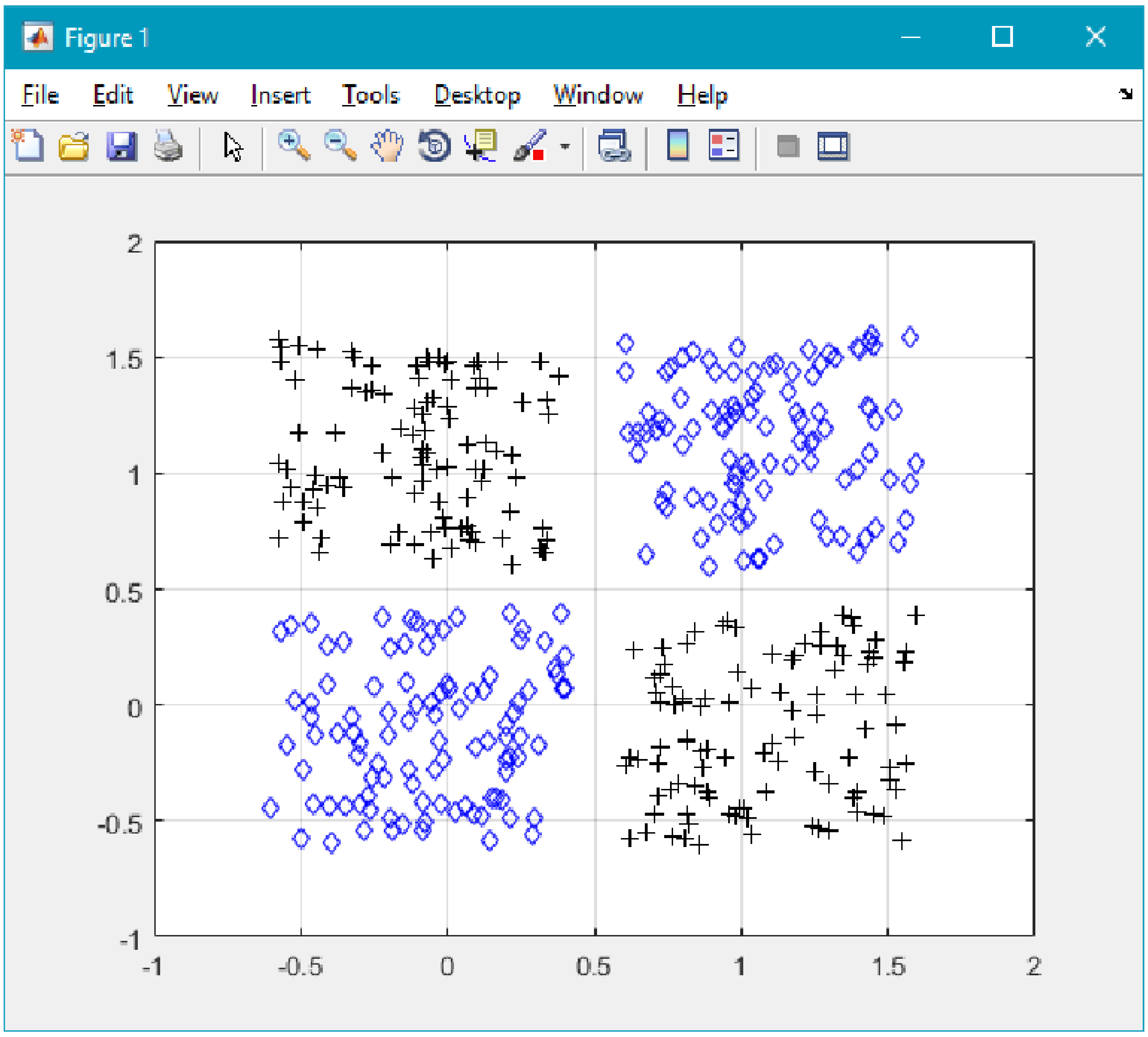


Рисунок 5.22 – Входные данные для двух классов, которые нужно разделить

Net = feedforwardnet([5 3]) – создаём сеть прямого распространения с двумя слоями, в первом будет 5 нейронов, во втором – 3. Структура сети показана на рисунке 3.23.

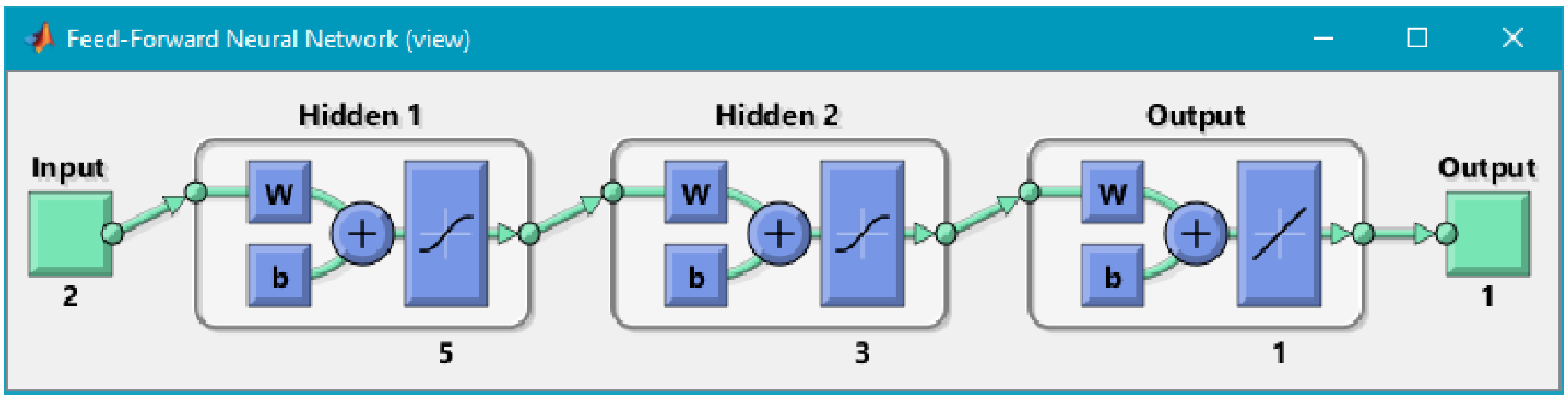


Рисунок 5.23 – Общая структура сети

Обратим внимание, что выходной нейрон имеет линейную функцию активации.

Все паттерны войдут в обучающую выборку, поэтому **net.divideParam.trainRatio = 1**, валидационную и тестовую выборку установим в 0.

Обучение сети осуществляется через **[net, tr, Y, E] = train(net, P, T)** – где **tr** – переменная, содержащая информацию об обучении (в среде Matlab её можно открыть и посмотреть на входящие в неё поля и их значения). Процесс обучения показан на рисунке 5.24, кривая обучения на рисунке 5.25.

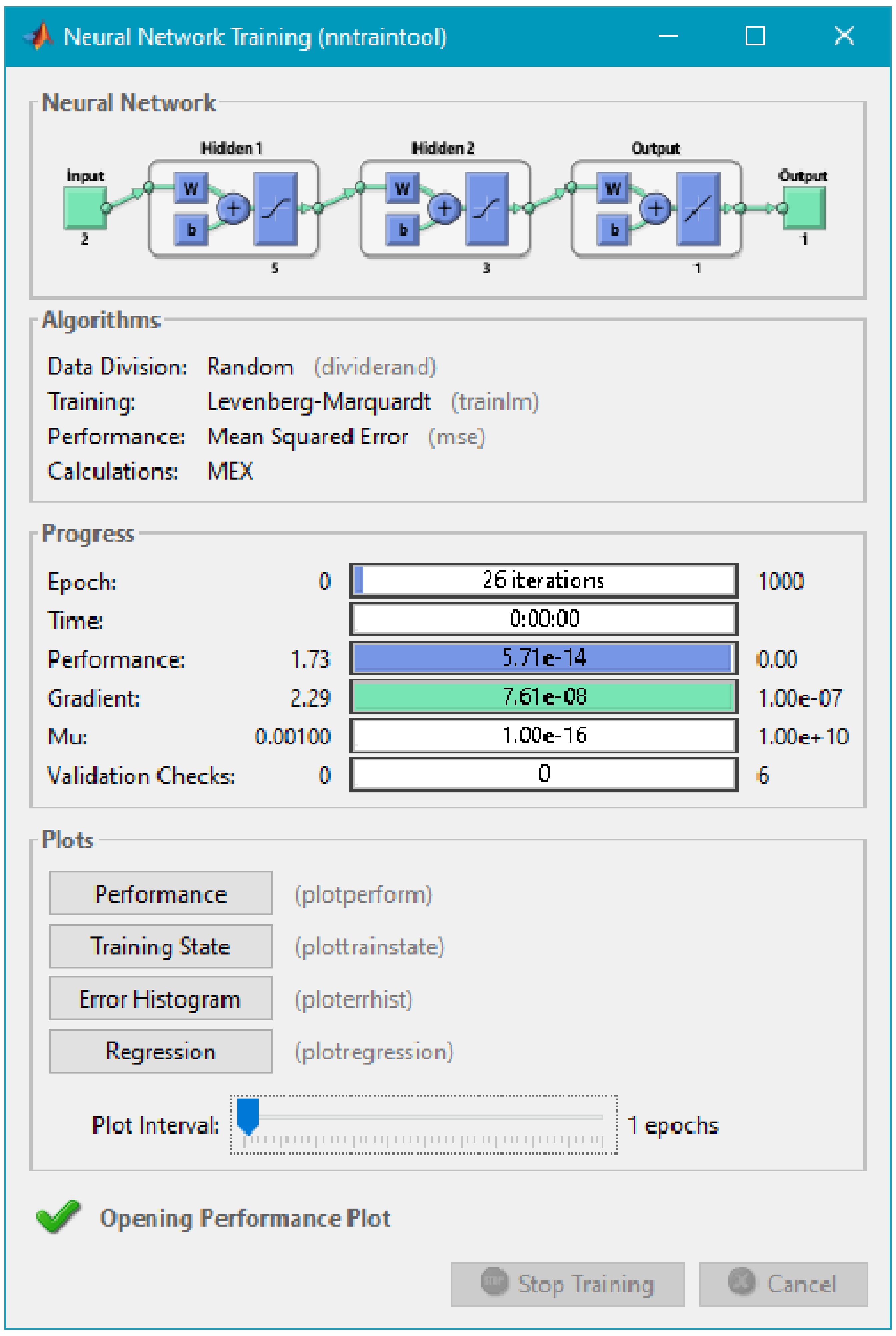


Рисунок 5.24 – Процесс обучения сети, потребовалось 26 итераций

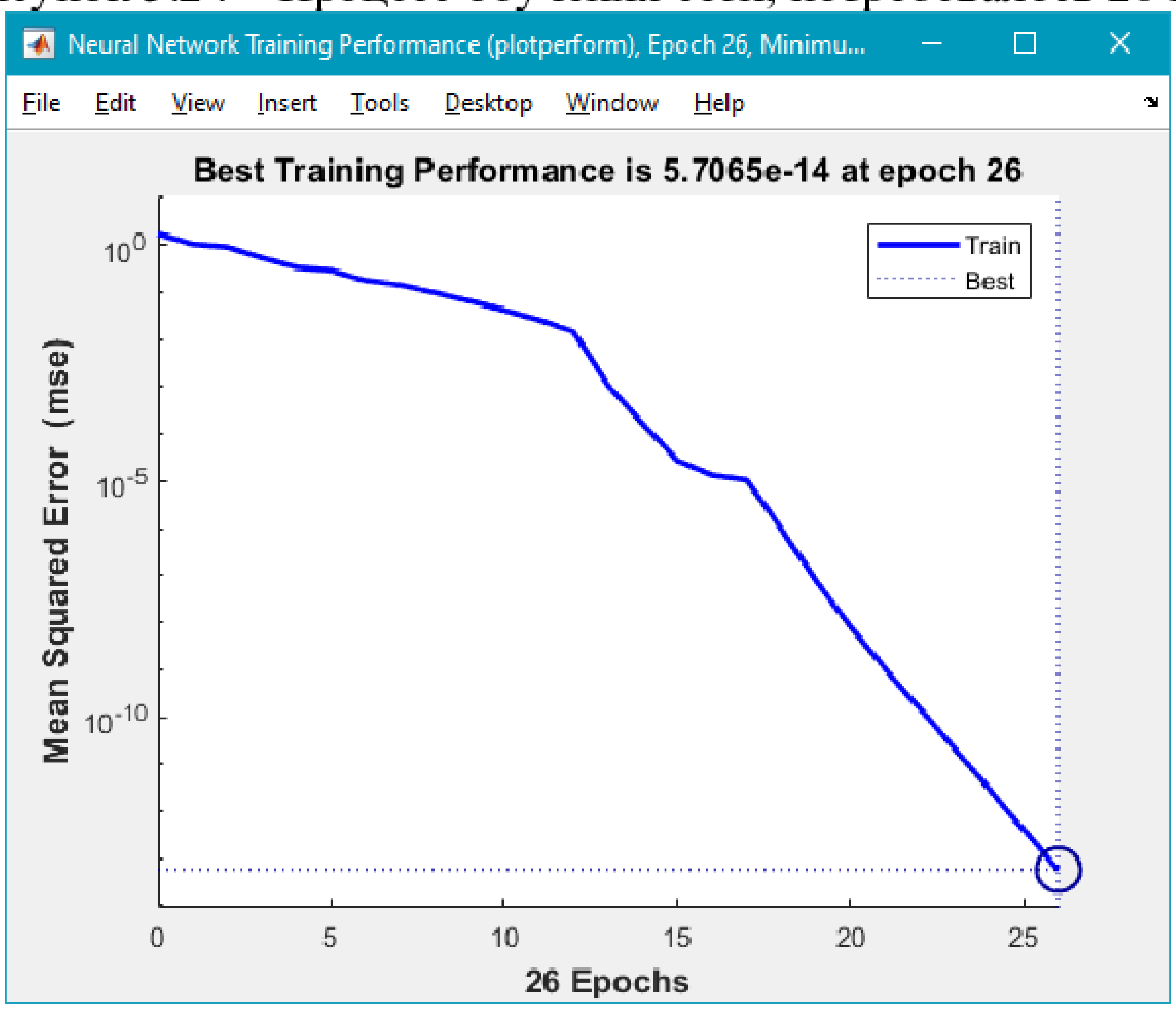


Рисунок 5.25 – Кривая обучения (MSE), лучший показатель меньше 10^{-10}

Далее отображаем на новой канве ожидаемый ответ сети, рисунок 5.26.

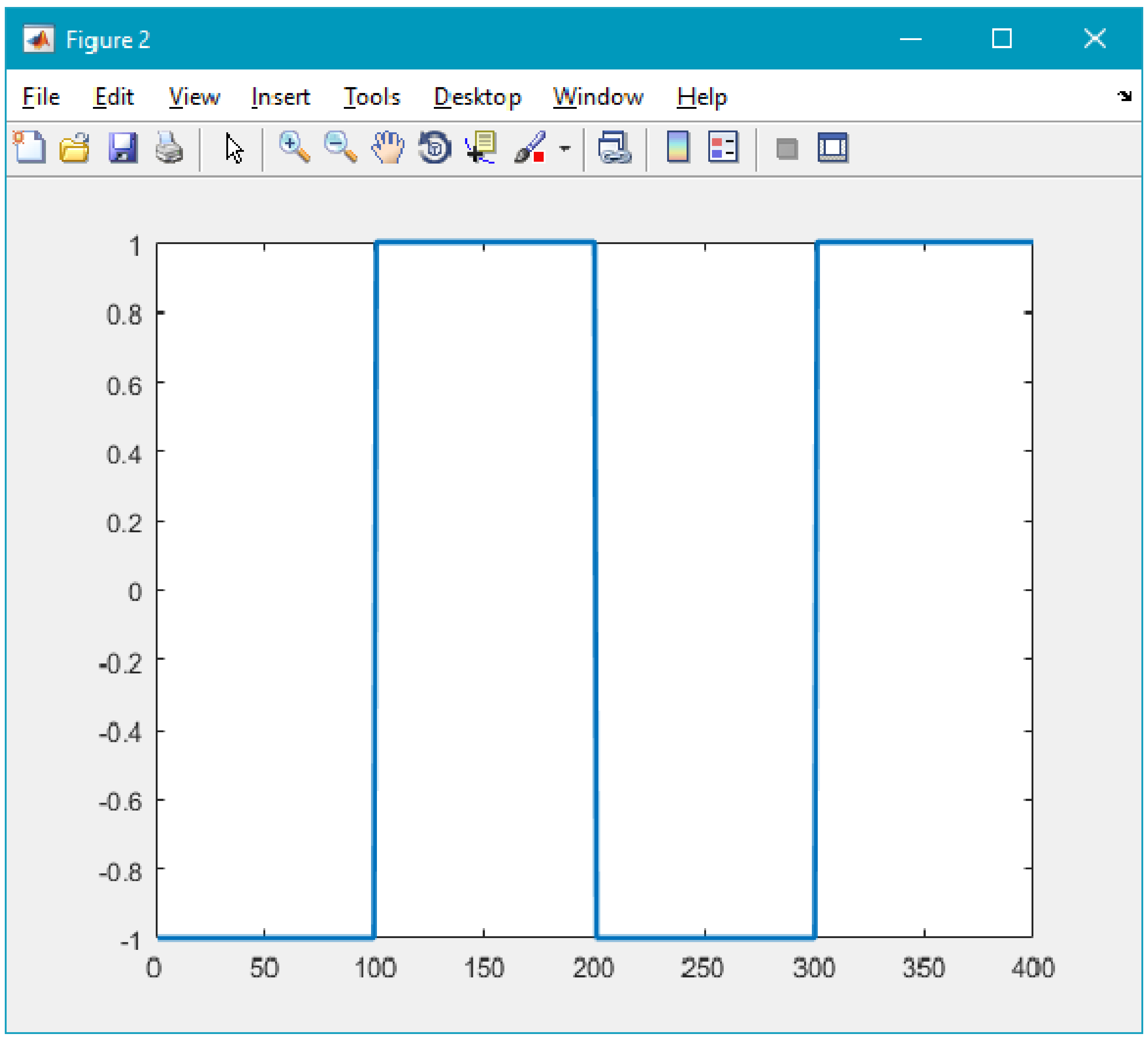


Рисунок 5.26 – Ожидаемый ответ сети (классы кодировались ±1)

На той же канве отображаем реальный ответ сети, рисунок 5.27. Видно, что графики совпадают.

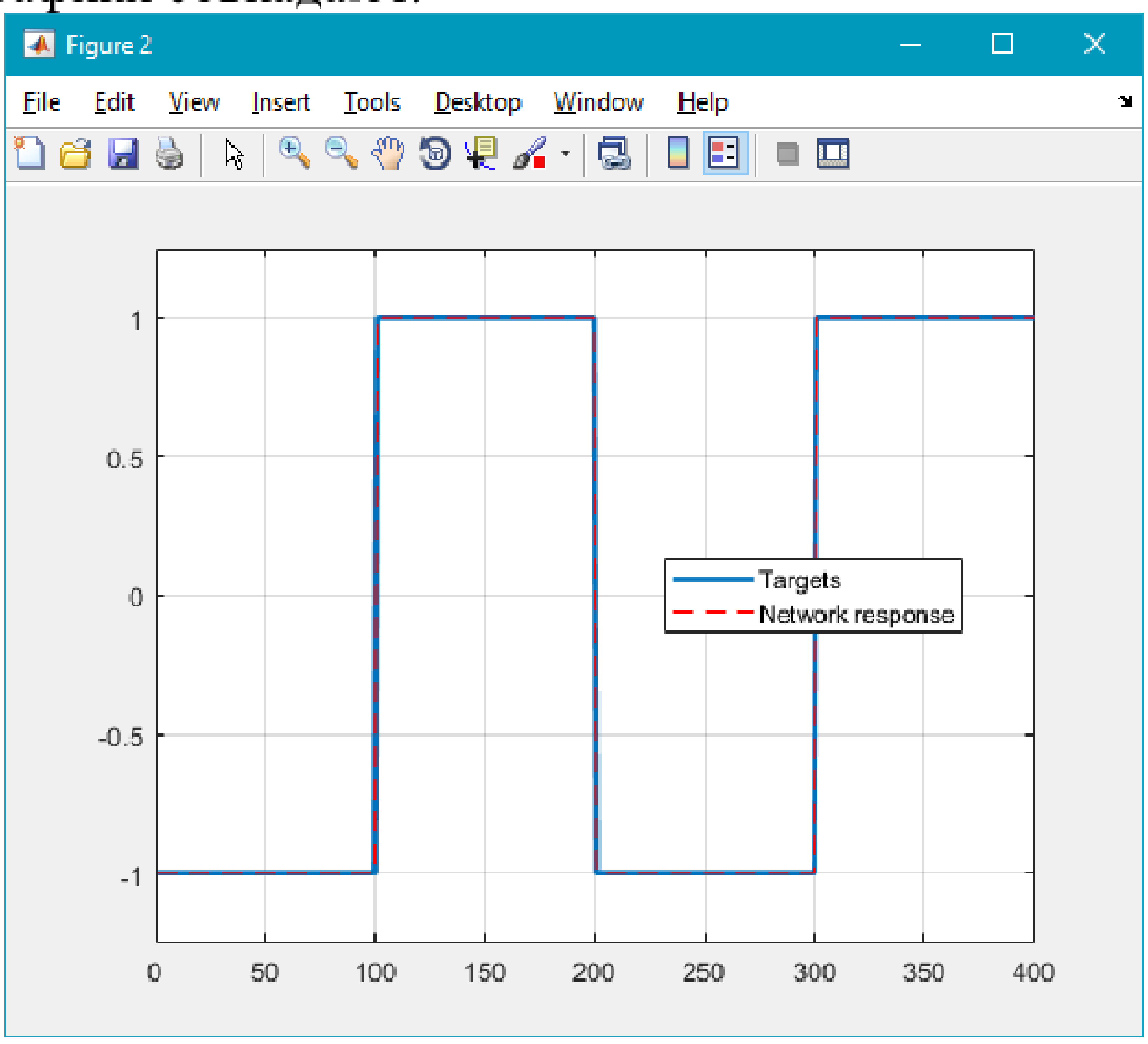


Рисунок 5.27 – Реальный ответ сети, графики совпадают

Ylim([-1.25 1.25]) – ограничивает ось OY данными значениями.

Чтобы отобразить получившееся решение уже нельзя просто построить прямые линии (как в случае с персептроном). Для этого нужно поверх первой канвы отобразить значения выходов сети на очень мелкой сетке (входные данные) и закрасить получившиеся ответы. Граница решения для разбиения двух классов окажется видна, рисунок 5.28.

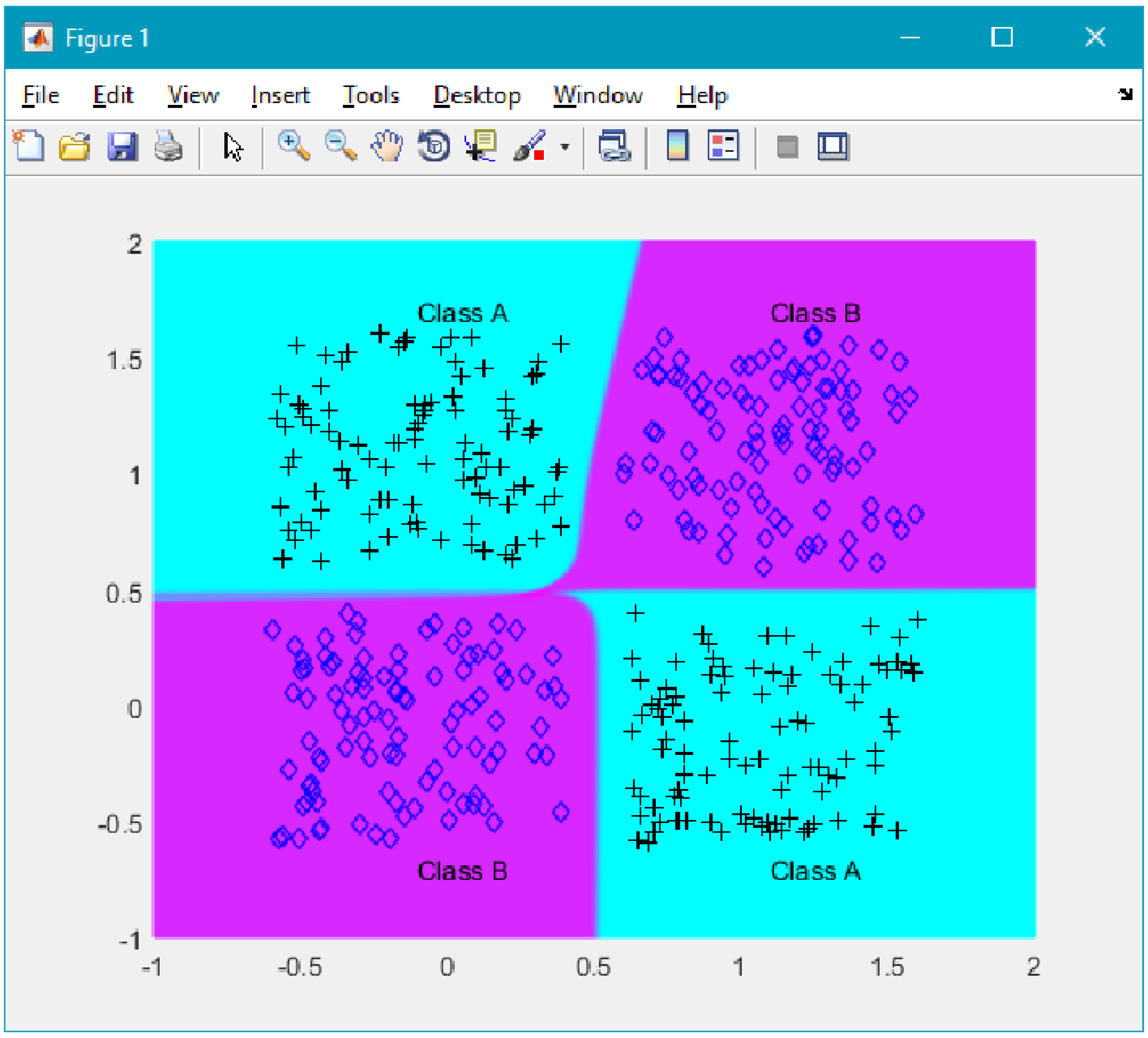


Рисунок 5.28 – Получившаяся граница решения для задачи XOR

Примечание. Сеть со структурой 2-5-3-1 явно избыточна для решения данной задачи.

Пример 6. Переопределение структуры сети.

Можно переопределить структуру сети, как было показано в начале данной лабораторной работы:

```
net.layers{1}.size = 5;
% Получаем двухслойную сеть, где скрытый слой имеет 5 нейронов
net = configure(net, P, T);
% Меняем функцию активацию на выходном нейроне на нелинейную
net.layers{2}.transferFcn = 'tansig';
net.divideParam.trainRatio = 1;
net.divideParam.valRatio = 0;
net.divideParam.testRatio = 0;
[net, tr, Y, E] = train(net, P, T);
```

Структура такой сети показана на рисунке 5.29.

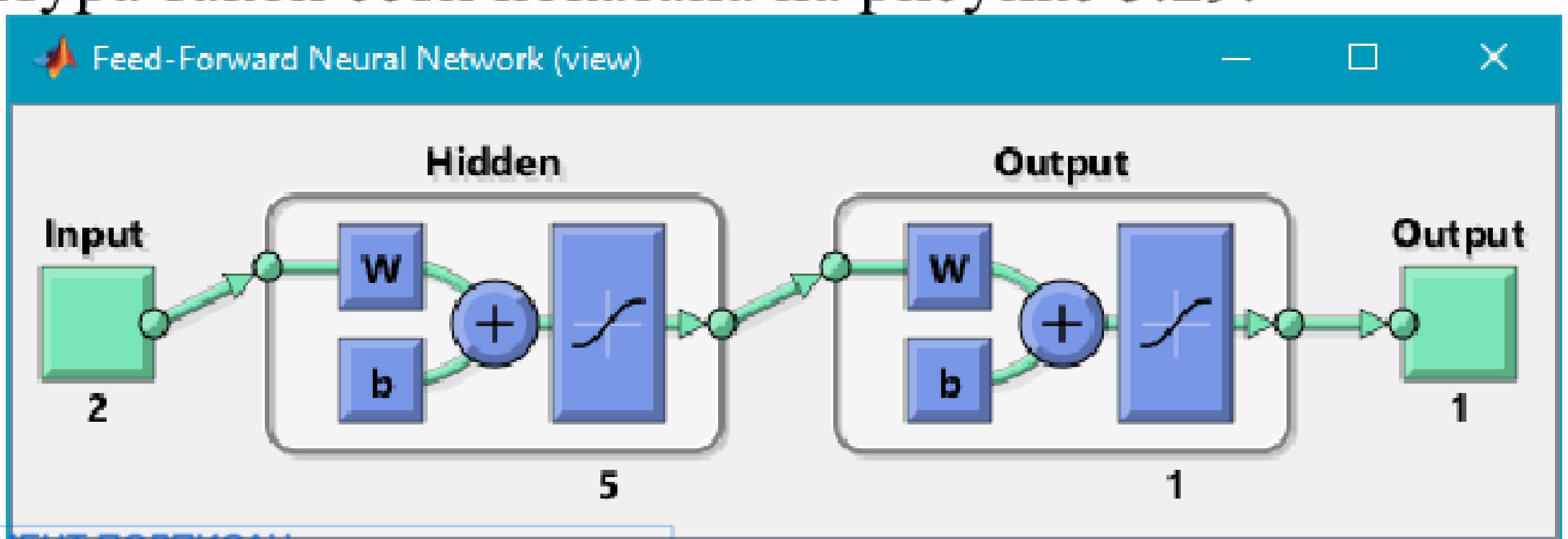


Рисунок 5.29 – Данная сеть тоже справится с задачей XOR

Однако, как видно из рисунка 5.30, ошибка обучения опустится в район 10^{-9}, что больше, чем у первой сети, что в целом хуже.

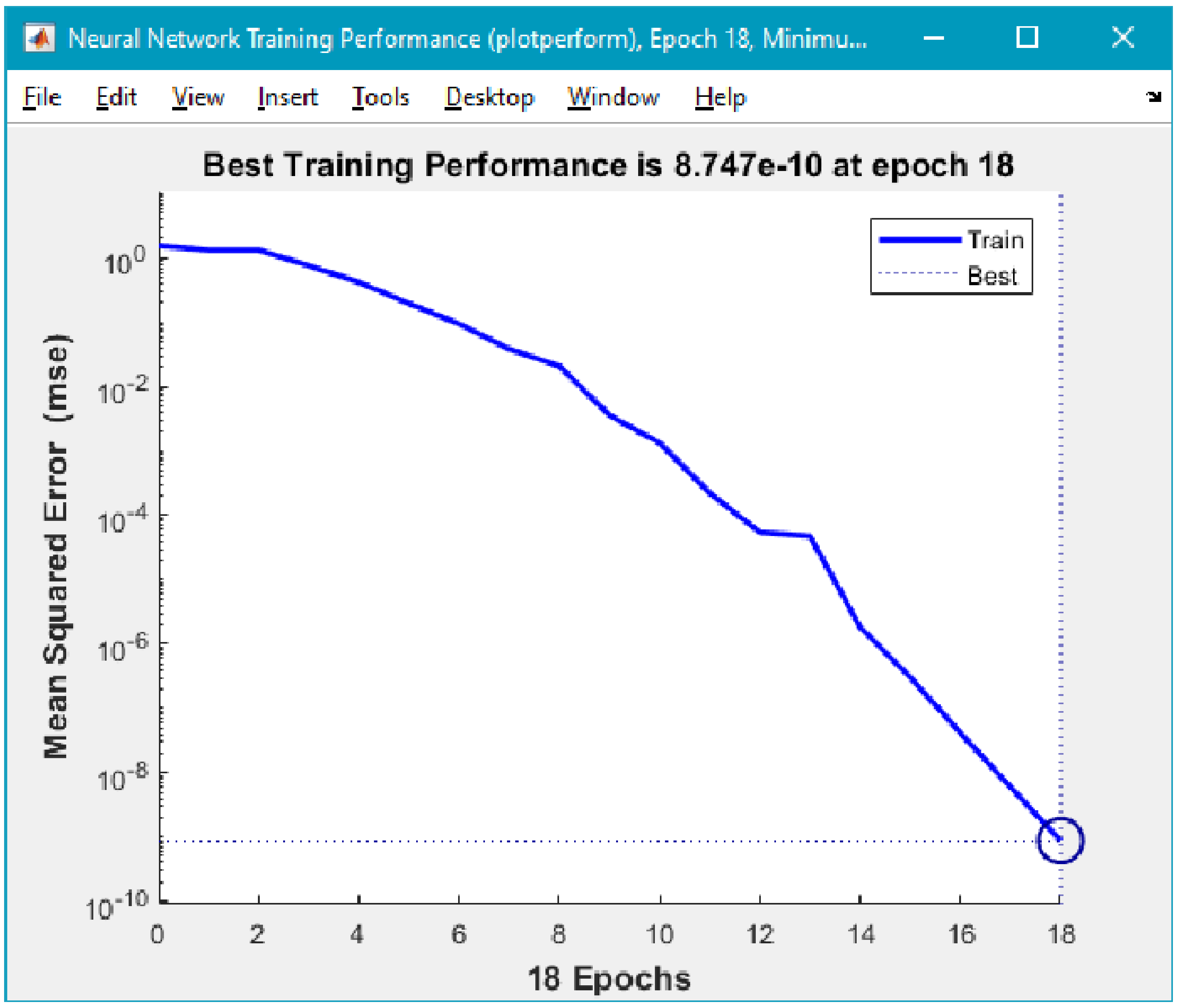


Рисунок 5.30 – Ошибка обучения для сети 2-5-1

Граница решений для этой сети показана на рисунке 5.31.

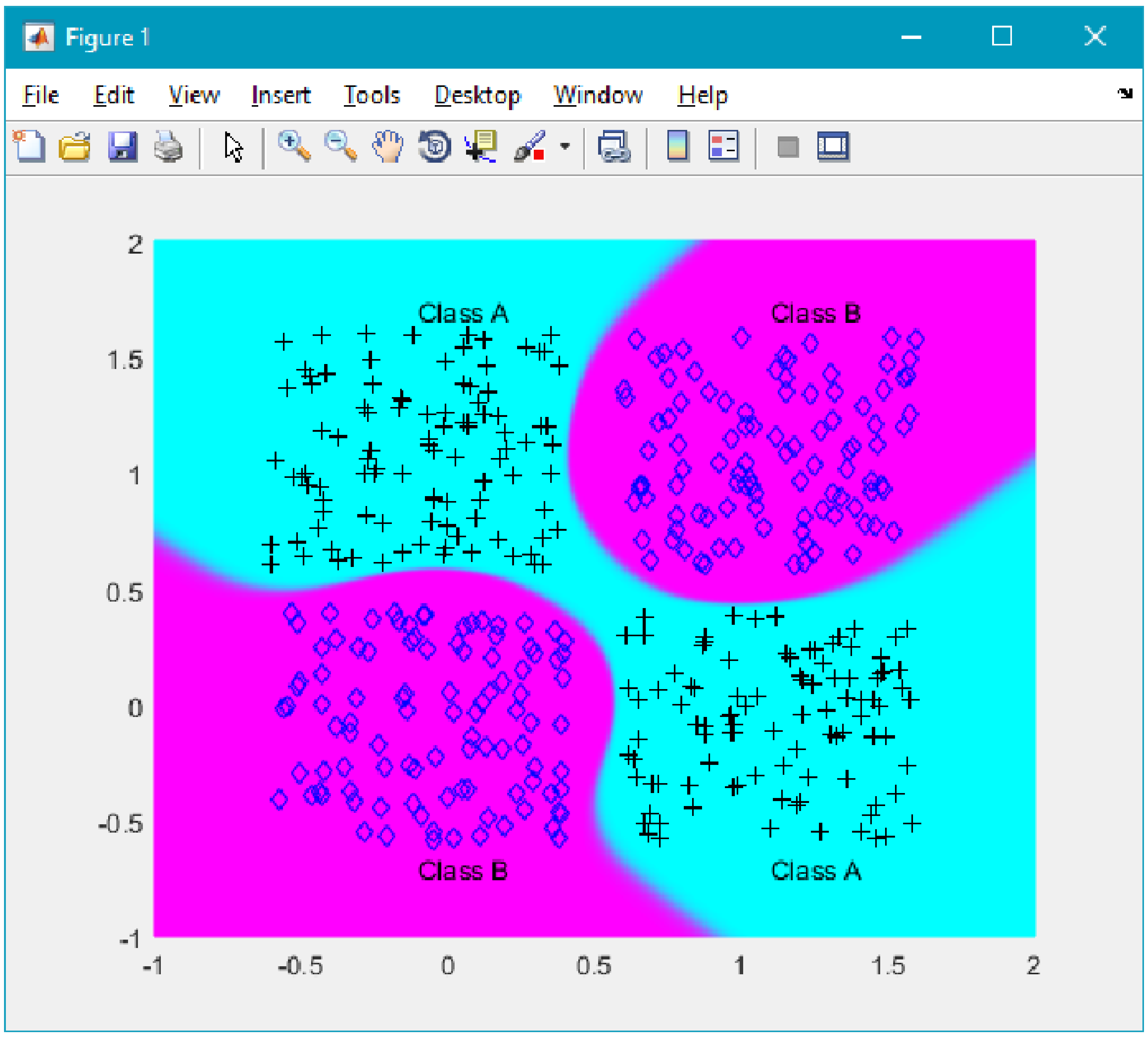


Рисунок 5. 31 – Граница решений для сети с пятью скрытыми нейронами

Сеть со структурой 2-3-1 даст ошибку, показанную на рисунке 5.32.

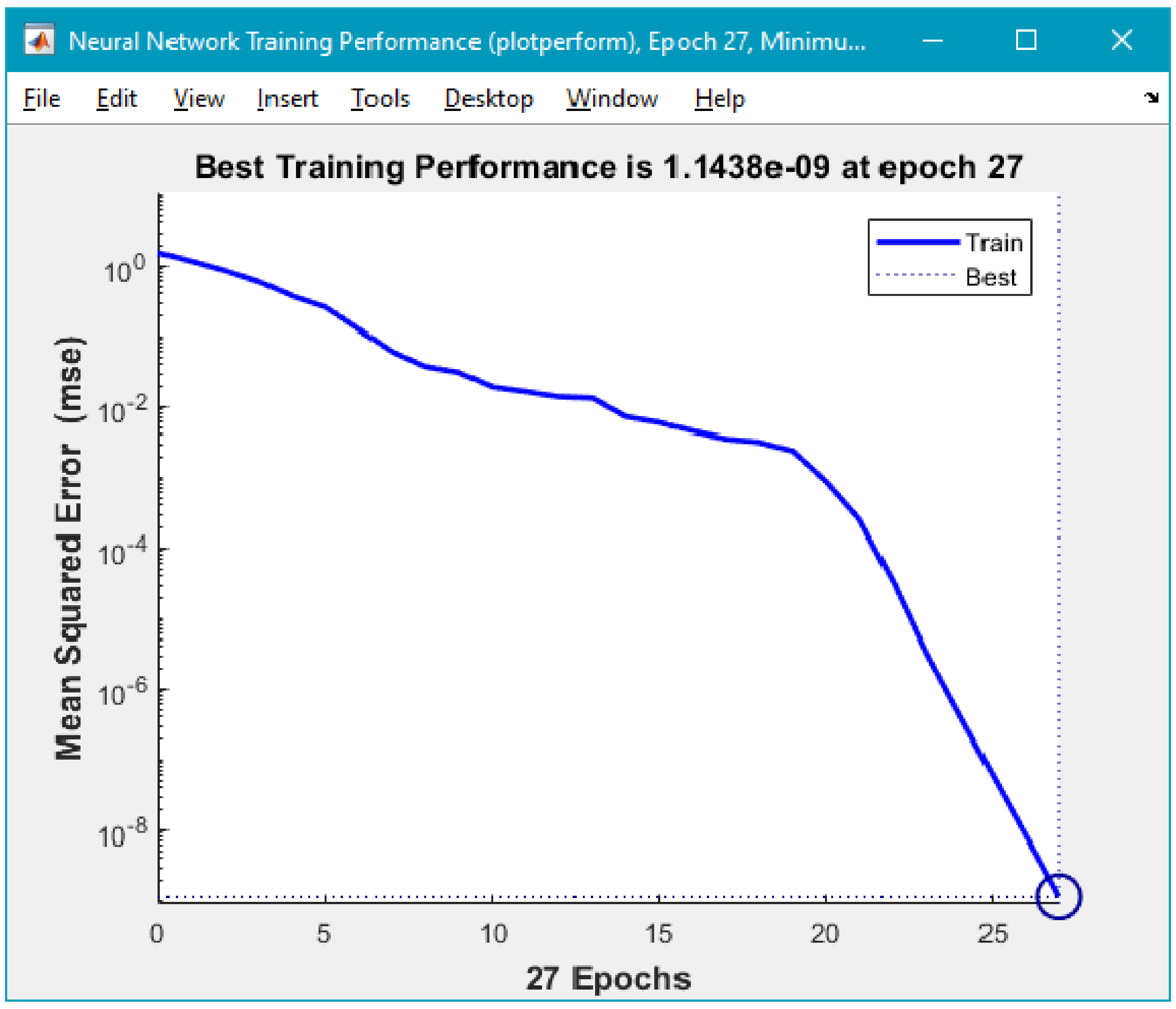


Рисунок 5.32 – Ошибка обучения для сети 2-3-1

Получившаяся граница решений для сети с тремя скрытыми нейронами показана на рисунке 5.33.

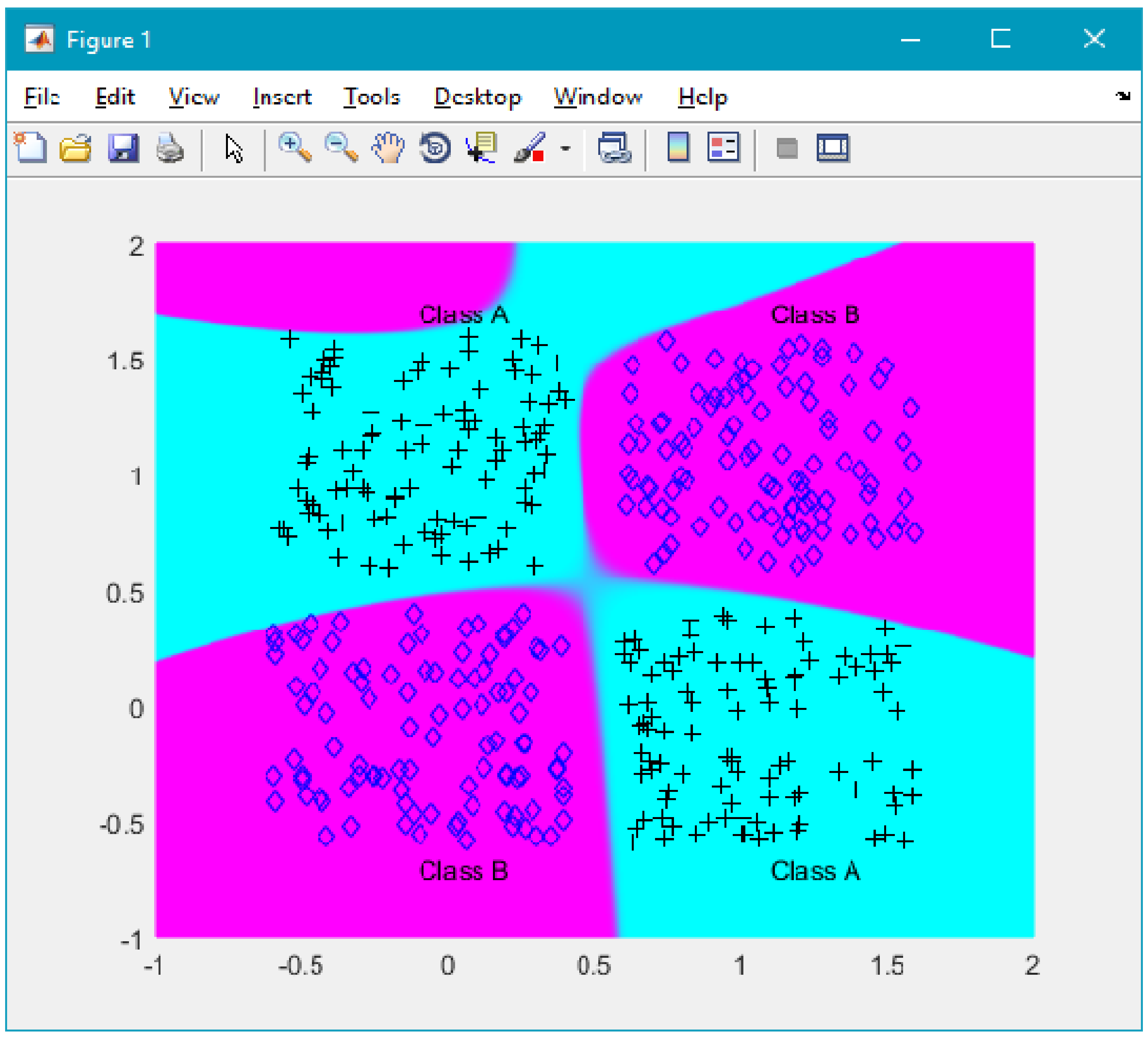


Рисунок 5.33 – Граница решений для сети из трёх нейронов в скрытом слое

Можно сделать вывод, что чем меньше количество нейронов в скрытом слое, тем грубее будут разделены классы и граница решений может в некоторых местах принимать хаотический вид (рисунок 5.33).

Конечно, такой разброс в решениях получается вследствие лёгкой разделимости классов.

Пример 7. Применение сети прямого распространения к разделению 4-х классов.

Теперь применим сеть прямого распространения к разделению 4-х классов (ранее решали эту задачу с помощью персептрона).

```
close all, clear all, clc, format compact
K = 100;
q = .6;
A = [rand(1, K)-q; rand(1, K)+q];
B = [rand(1, K)+q; rand(1, K)+q];
C = [rand(1, K)+q; rand(1, K)-q];
D = [rand(1, K)-q; rand(1, K)-q];
figure(1);
plot(A(1, :), A(2, :), 'k+');
hold on;
grid on;
plot(B(1, :), B(2, :), 'b*');
plot(C(1, :), C(2, :), 'kx');
plot(D(1, :), D(2, :), 'bd');
text(.5-q, .5+2*q, 'Class A');
text(.5+q, .5+2*q, 'Class B');
text(.5+q, .5-2*q, 'Class C');
text(.5-q, .5-2*q, 'Class D');
% Кодируем классы
a = [-1 -1 -1 +1]';
b = [-1 -1 +1 -1]';
c = [-1 +1 -1 -1]';
d = [+1 -1 -1 -1]';
P = [A B C D];
T = [repmat(a, 1, length(A)) repmat(b, 1, length(B)) repmat(c, 1, length(C)) repmat(d, 1, length(D))];
net =feedforwardnet([4 3]);
net.divideParam.trainRatio = 1;
net.divideParam.valRatio = 0;
net.divideParam.testRatio = 0;
[net, tr, Y, E] = train(net, P, T);
view(net);
% Получаем в "m" все максимальные элементы по столбцам, а в "i"
% их строковые индексы
[m, i] = max(T);
% Получаем в j индексы строк максимальных элементов
[m, j] = max(Y);
N = length(Y);
k = 0;
% Сравниваем эти индексы, они должны совпасть, если они не
% совпадают, то увеличиваем k на 1, т.е. k будет содержать количество
% ответов ИНС, которые не совпадают с метками
if find(i - j),
k = length(find(i-j));
end
% Высчитываем это количество в процентах
fprintf('Correct classified samples: %.if%% samples\n', 100*(N-k)/N);
figure;
% Строим верхний график на канве
subplot(211);
plot(T');
```

```
title('Targets');
ylim([-2 2]);
grid on;
% Строим нижний график на канве
subplot(212);
plot(Y');
title('Network response');
xlabel('# sample');
ylim([-2 2]);
grid on;
span = -1:.01:2;
[P1, P2] = meshgrid(span, span);
pp = [P1(:) P2(:)]';
aa = net(pp);
figure(1);
% Наносим на сетку все ответы первого нейрона для
% всей выборки, они должны занять ¼ от площади всех классов
m = mesh(P1, P2, reshape(aa(1, :), length(span), length(span))-5);
set(m, 'facecolor', [1 0.2 .7], 'linestyle', 'none');
hold on;
% Тоже самое делаем для других нейронов
m = mesh(P1, P2, reshape(aa(2, :), length(span), length(span))-5);
set(m, 'facecolor', [1 1.0 0.5], 'linestyle', 'none');
m = mesh(P1, P2, reshape(aa(3, :), length(span), length(span))-5);
set(m, 'facecolor', [.4 1.0 0.9], 'linestyle', 'none');
m = mesh(P1, P2, reshape(aa(4, :), length(span), length(span))-5);
set(m, 'facecolor', [.3 .4 0.5], 'linestyle', 'none');
view(2);
```

Создаём 4 класса, каждый представлен 100 паттернами, рисунок 5.34.

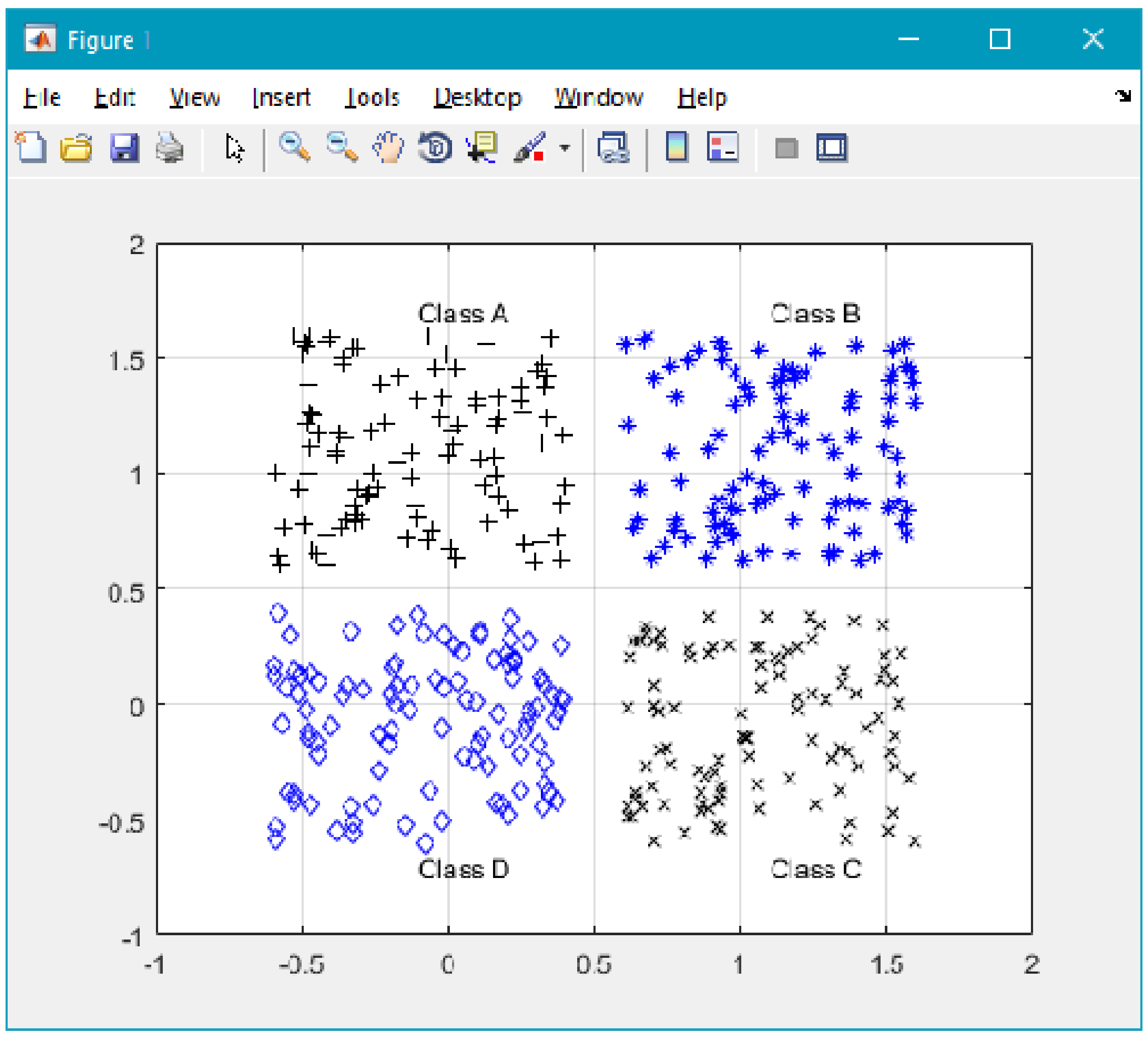


Рисунок 5.34 – Четыре класса, которые нужно разделить

Далее создаём сеть со структурой 2-4-3-4 и обучаем её, рисунки 3.35 – 3.37.

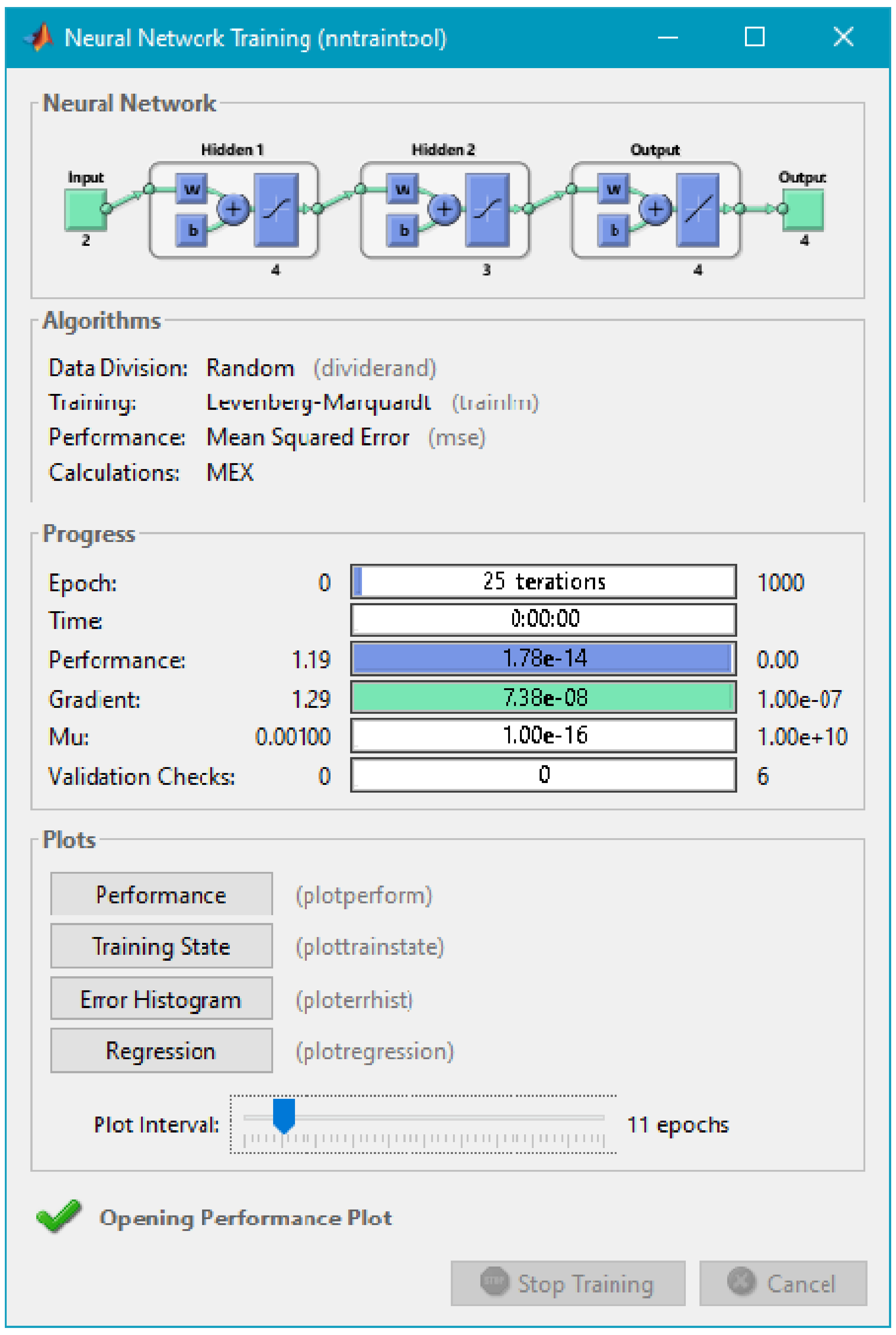


Рисунок 5.35 – Обучение сети, потребовалось 25 итерации

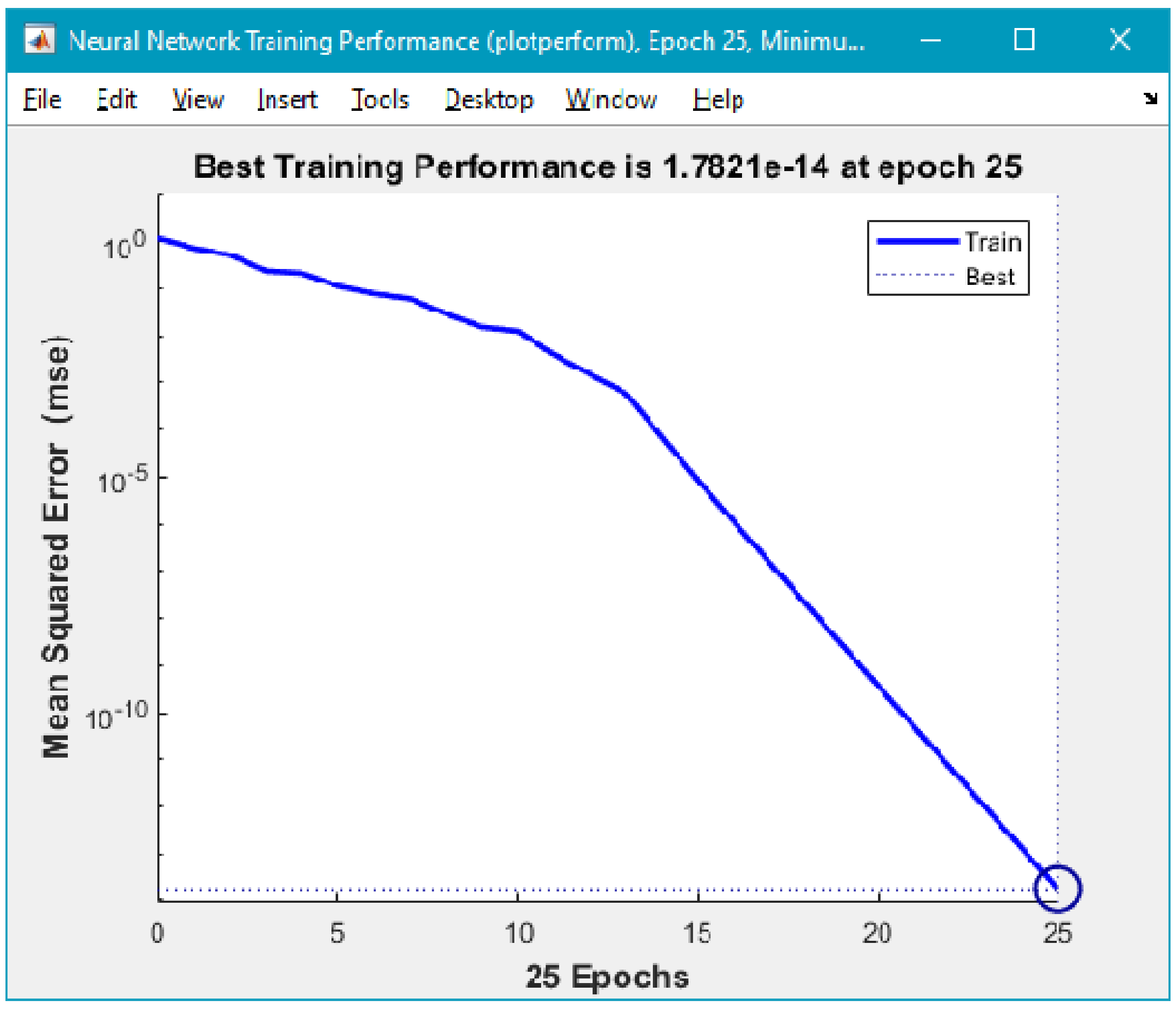


Рисунок 5.36 – Ошибка обучения очень хорошая, стала ниже 10^{-10} на четыре порядка

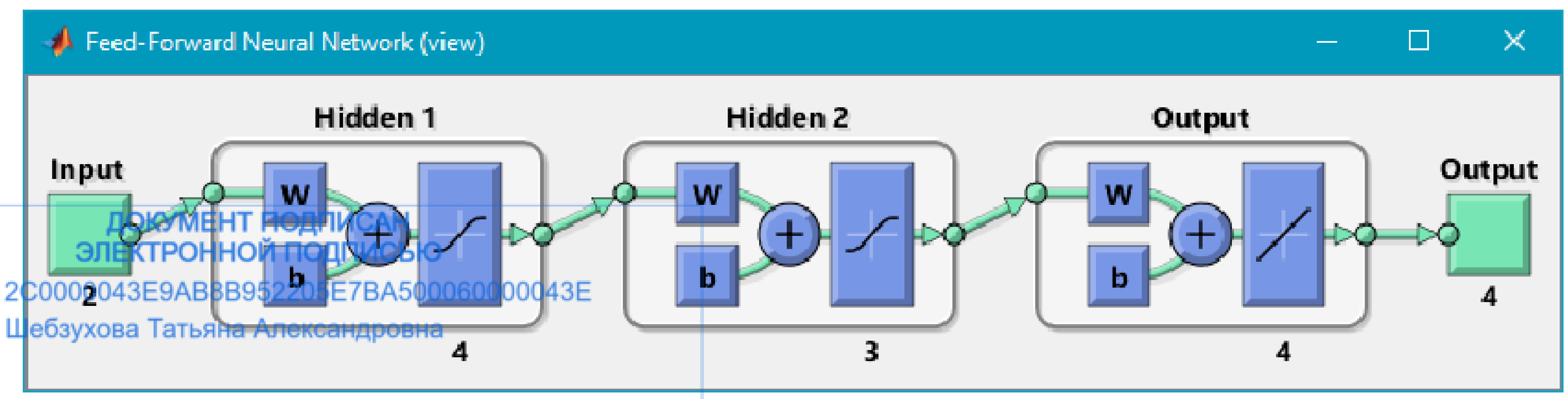


Рисунок 5.37 – Общая структура сети, которая использовалась

Разумеется, предложенная структура не единственная, на самом деле хватило бы сети с одним скрытым слоем и четырьмя нелинейными нейронами на выходном слое.

Метки и ответы сети изображены на рисунке 5.38, видно, что графики полностью совпадают, т.е. сеть точно моделирует все реакции на входные значения.

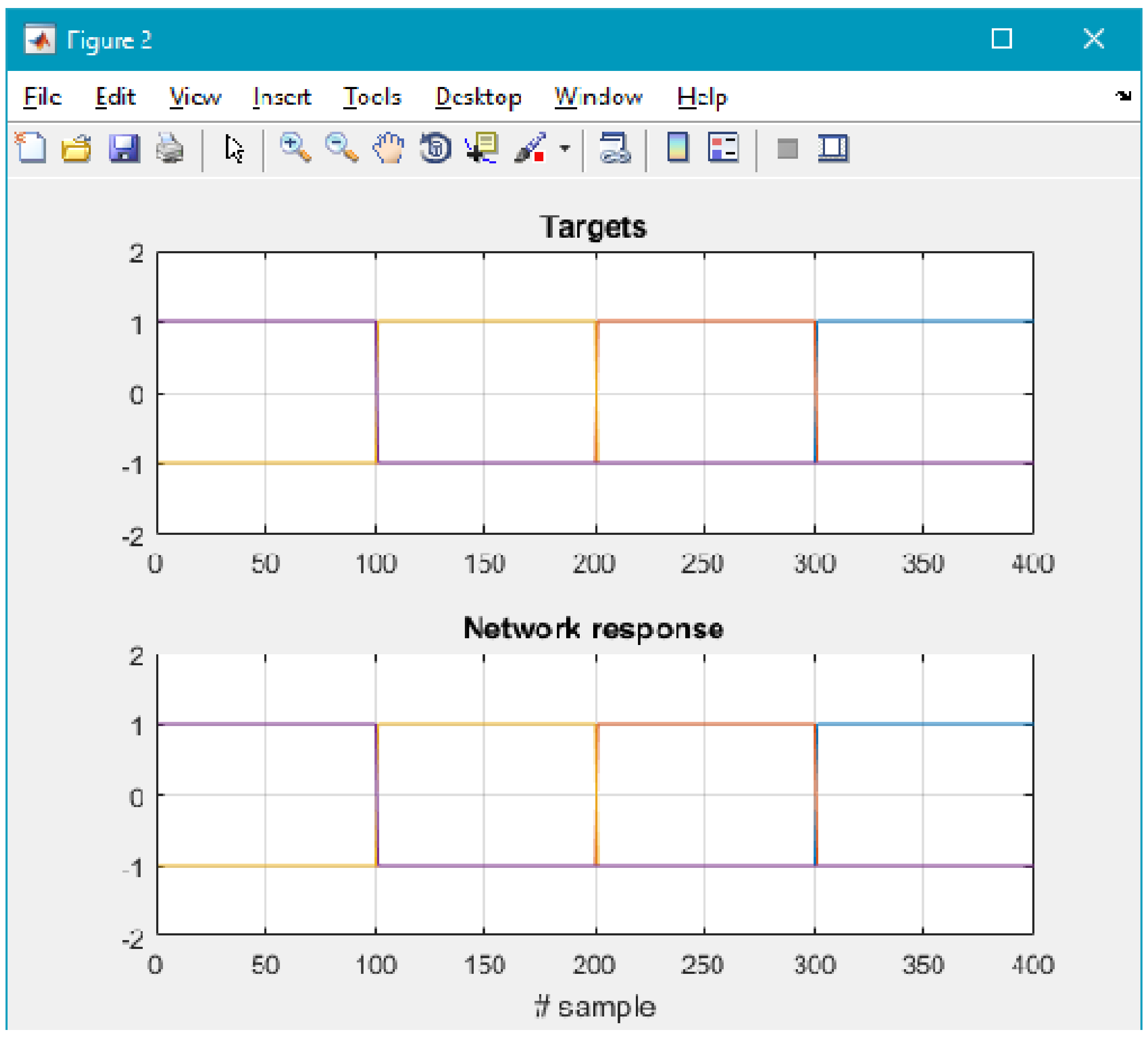


Рисунок 5.38 – График ожидаемых и реальных ответов ИНС

После построения границ решения можно видеть, что каждый класс отделён от другого, рисунок 5.39.

Если сравнить это решение с тем, которое было достигнуто с помощью персептрона, то станет очевидным, что возможности по отделению одних классов от других у ИНС прямого распространения значительно выше, чем у персептрона, т.к. такие сети позволяют строить более сложные границы.

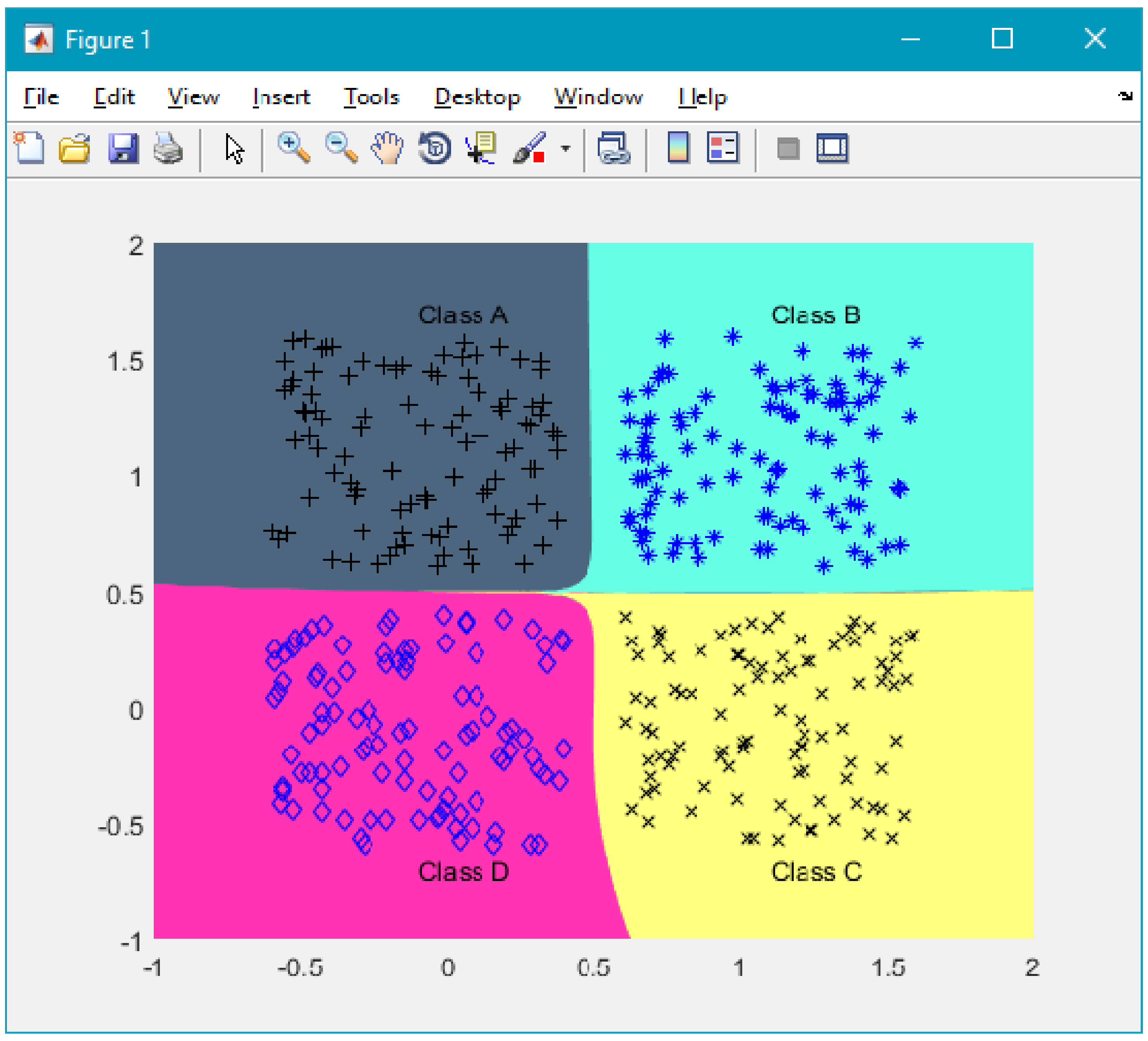


Рисунок 5.39 – Границы решения для задачи по разделению четырёх классов с помощью сети прямого распространения

Аппаратура и материалы. Для выполнения лабораторной работы необходимо использовать следующее: аппаратное обеспечение: персональный компьютер и программное обеспечение: операционную систему Windows 10 и выше; Microsoft Office13 и выше, системы компьютерной математики Machcad и MATLAB R2017a и выше.

Указание по технике безопасности. Самостоятельно не производить: установку и удаление программного обеспечения; ремонт персонального компьютера. Соблюдать правила технической эксплуатации и техники безопасности при работе с электрооборудованием.

Методика и порядок выполнения работы

Выполните предложенные задания, предварительно рассмотрев и выполнив все приведенные примеры.

Задание 5.1. В индивидуальном варианте (таблица 5.1) дано расположение двух классов (А и В) по «углам» квадрата (по типу как на рисунке 5.39). Каждый «угол» представлен 200 точками. Требуется отделить класс А от В с помощью персептрона и полносвязанной многослойной сети прямого распространения.

Таблица 3.1 – Варианты заданий

Номер варианта	Условие задачи	Номер варианта	Условие задачи
1	$\begin{vmatrix} B & A \\ B & B \end{vmatrix}$	8	$\begin{vmatrix} B & A \\ B & A \end{vmatrix}$
2	$\begin{vmatrix} A & B \\ B & B \end{vmatrix}$	9	$\begin{vmatrix} A & B \\ B & A \end{vmatrix}$

Номер варианта	Условие задачи	Номер варианта	Условие задачи
3	$\begin{pmatrix} B & B \\ A & B \end{pmatrix}$	10	$\begin{pmatrix} B & A \\ A & B \end{pmatrix}$
4	$\begin{pmatrix} B & B \\ B & A \end{pmatrix}$	11	$\begin{pmatrix} A & A \\ B & A \end{pmatrix}$
5	$\begin{pmatrix} A & A \\ B & B \end{pmatrix}$	12	$\begin{pmatrix} B & A \\ A & A \end{pmatrix}$
6	$\begin{pmatrix} A & B \\ A & B \end{pmatrix}$	13	$\begin{pmatrix} A & B \\ A & A \end{pmatrix}$
7	$\begin{pmatrix} B & B \\ A & A \end{pmatrix}$	14	$\begin{pmatrix} A & A \\ A & B \end{pmatrix}$

Содержание отчета и его форма

Подготовьте отчет, в котором приведите технологию выполнения заданий.

Отчет по лабораторной работе должен содержать:

1) название работы;
2) цель лабораторной работы;
3) формулировку задания и технологию его выполнения;
4) ответы на контрольные вопросы;
5) приложение – файлы выполненных заданий.

Вопросы для защиты работы

1. Как кодируются классы для персептронов или сетей прямого распространения?
2. Опишите последовательность действий для нанесения входных значений на канву, чтобы визуально было видно расположение классов относительно друг друга.
3. Опишите последовательность действий для построения на канве границ решения задачи классификации.
4. Приведите пример сети прямого распространения с одним скрытым слоем, с помощью которой можно решить задачу о разделении 4-х классов в рассмотренной лабораторной работе.
5. Почему сеть прямого распространения может лучше справляться с задачами классификации, чем персептрон?

ЛАБОРАТОРНАЯ РАБОТА 6
Оптимизационные модели принятия решений. Постановка и решение задач оптимизации

Цель и содержание: Построение моделей задач и изучение методов их решения. Решение задачи принятия решений в условиях определенности.

Организационная форма занятий: решение проблемных задач, разбор конкретных ситуаций

Вопросы для обсуждения на лабораторном занятии: построение моделей оптимизационных задач принятия решений в условиях определенности и их решение современными средствами информационных технологий.

Формируемые компетенции: ОПК-7.1 ОПК-7.2 ОПК-8.1 ОПК-8.2

Теоретическое обоснование

Изучите теоретический материал по данной теме.

Рассмотрим информационные модели принятия решений в условиях определенности на основе задач линейного программирования

Несмотря на требование линейности целевых функций (критериев оптимальности) и ограничений в линейном программировании решаются многочисленные задачи: распределения ресурсов; управления запасами; сетевого и календарного планирования; транспортные задачи, задачи теории игр и т.д.

Рассмотрим задачу на определение оптимального выпуска продукции. Имеются m видов ресурсов в количествах b_1, b_2, . , b_i, b_m и n видов изделий. Задана матрица $A=\|a_{ij}\|$, $i = 1,\ldots,m$, $j = 1,\ldots,n$, где a_{ij} характеризует нормы расхода i-го ресурса на единицу j-го вида изделий. Эффективность производства j-го вида изделий характеризуется показателем C_j, удовлетворяющим условию линейности. Нужно определить такой план выпуска изделий (оптимальный ассортимент), при котором суммарный показатель эффективности будет наибольший.

Обозначим количество единиц j-го вида изделий, выпускаемых предприятием, через x_j. Тогда математическая модель этой задачи будет иметь такой вид:

$$F(x)=\sum_{j=1}^{n} c_j x_j \quad \text{(max)}$$

при ограничениях:

$$\sum_{j=1}^{n} a_{ij} x_j \ge b_i, \quad j=\overline{1,n} \qquad (6.1)$$

$$x_j \ge 0,\ j=\overline{1,n}.$$

Рассмотрим решение задачи линейного программирования в Excel.

Пример 1. Найти максимальное значение целевой функции:

$$F(x)=50x_1+40x_2$$

при ограничениях:

$$\left| 2x_1+5x_2\le 20 ¿ \right| 8x_1+5x_2\le 40 ¿¿¿$$

$$x_1\ge 0,\ x_2\ge 0.$$

Решение. Для решения задачи необходимо выполнить следующие действия:

1. Заполнить рабочий лист следующим образом:

1.1. Ввести в ячейки (рисунок. 6.1):

А2 – формулу целевой функции, заменив x_1 и x_2 на С2 и С3: =E1*C2+F1*C3 или (=50*C1+40*C2). В первом случае в формуле указываются ссылки на ячейки, содержащие коэффициенты целевой функции, во втором случае просто вводятся их числовые значения;

А4 – формулу первого ограничения: =E3*C2+F3*C3 или (=2*C1+5*C2). Если ввести формулу, со ссылками на ячейки, указав какие из ячеек должны при копировании не меняться (для этого надо нажать на клавиатуре функциональную клавишу F4), то в ячейки А5 и А6 можно будет просто скопировать введенную формулу;

А5 – формулу второго ограничения: =E4*C2+F4*C3 или (=8*C1+5*C2);

А6 – формулу третьего ограничения: =E5*C2+F5*C3 или (=5*C2+6*C2);

А8 – ссылку на ячейку С2;

А9 – ссылку на ячейку С3.

1.2. Заполнить ячейки С2 и С3, положив начальные значения переменных равными нулю.

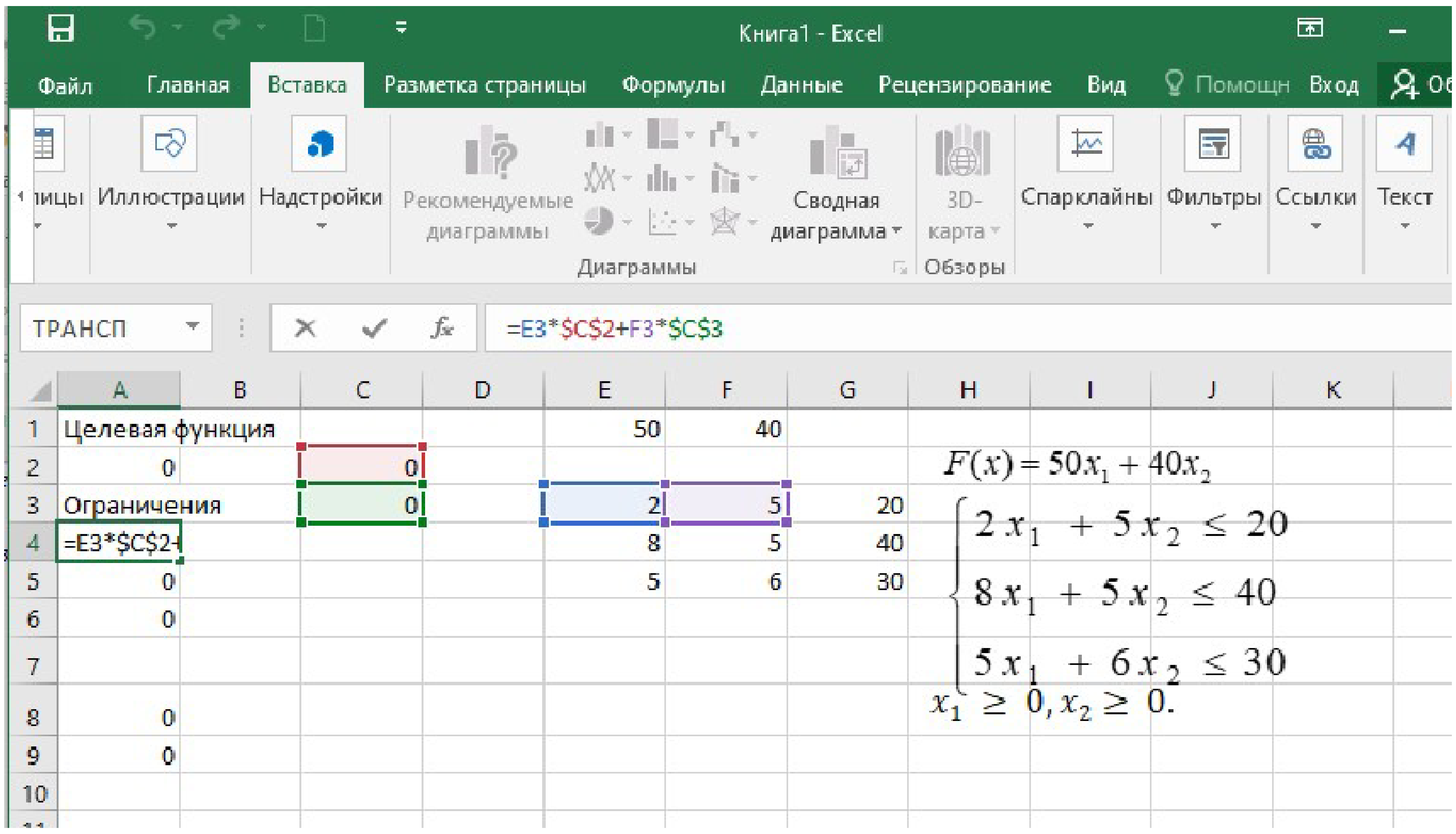


Рисунок 6.1 – Рабочий лист с условием задачи

2. Выделить ячейку, содержащую целевую функции, в нашем случае, это A2 и выбрать на вкладке **Данные**, в группе **Анализ**, надстройку **Поиск решений**. Появится окно диалога **Поиск решения** (рисунок 6.2), с помощью которого необходимо заполнить соответствующие поля:

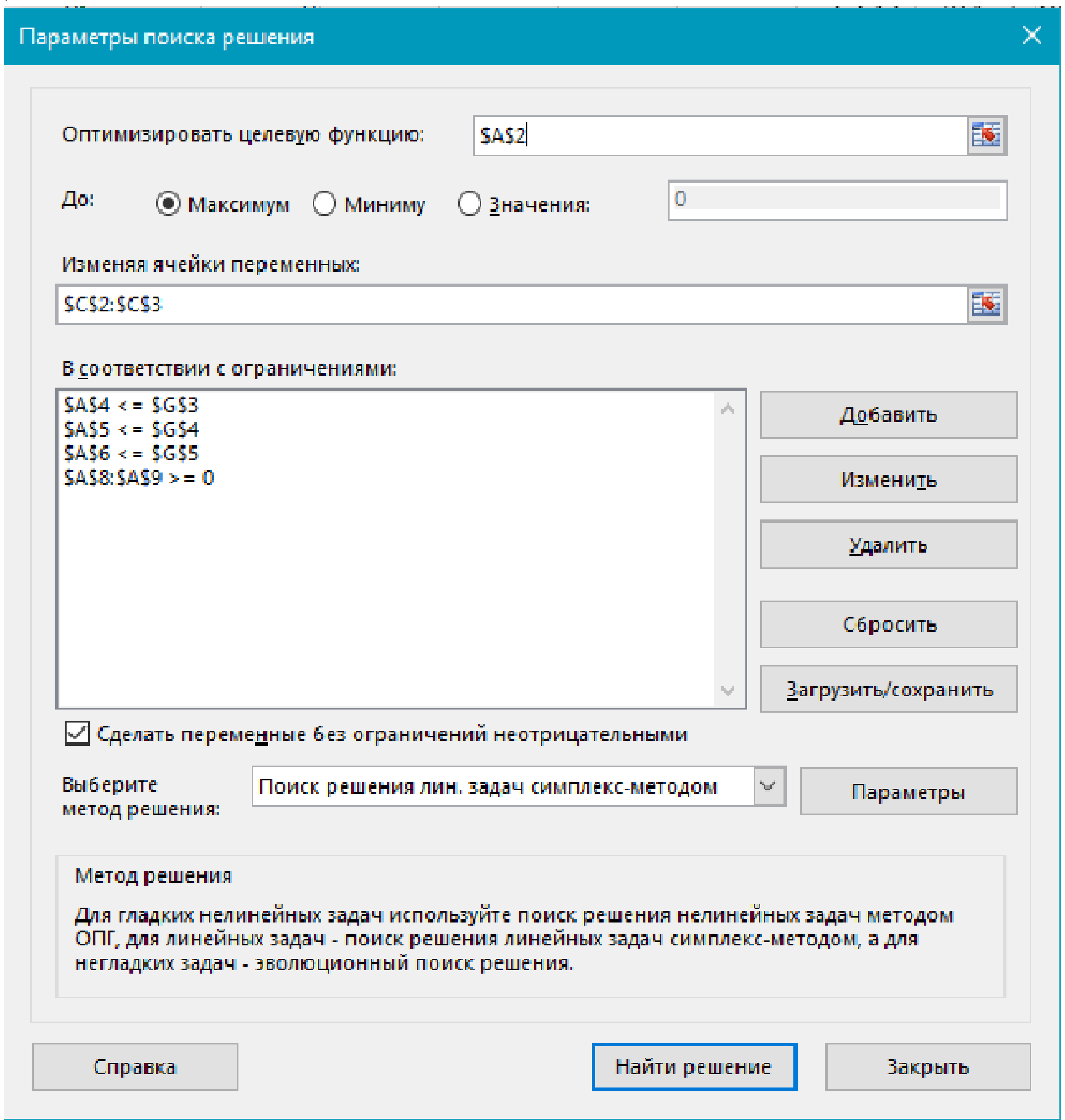


Рисунок 6.2 – Окно диалога **Поиск решения** с введенными данными

3. После окончания расчета Excel откроет окно диалога **Результаты поиска решения** (рисунок 6.3).

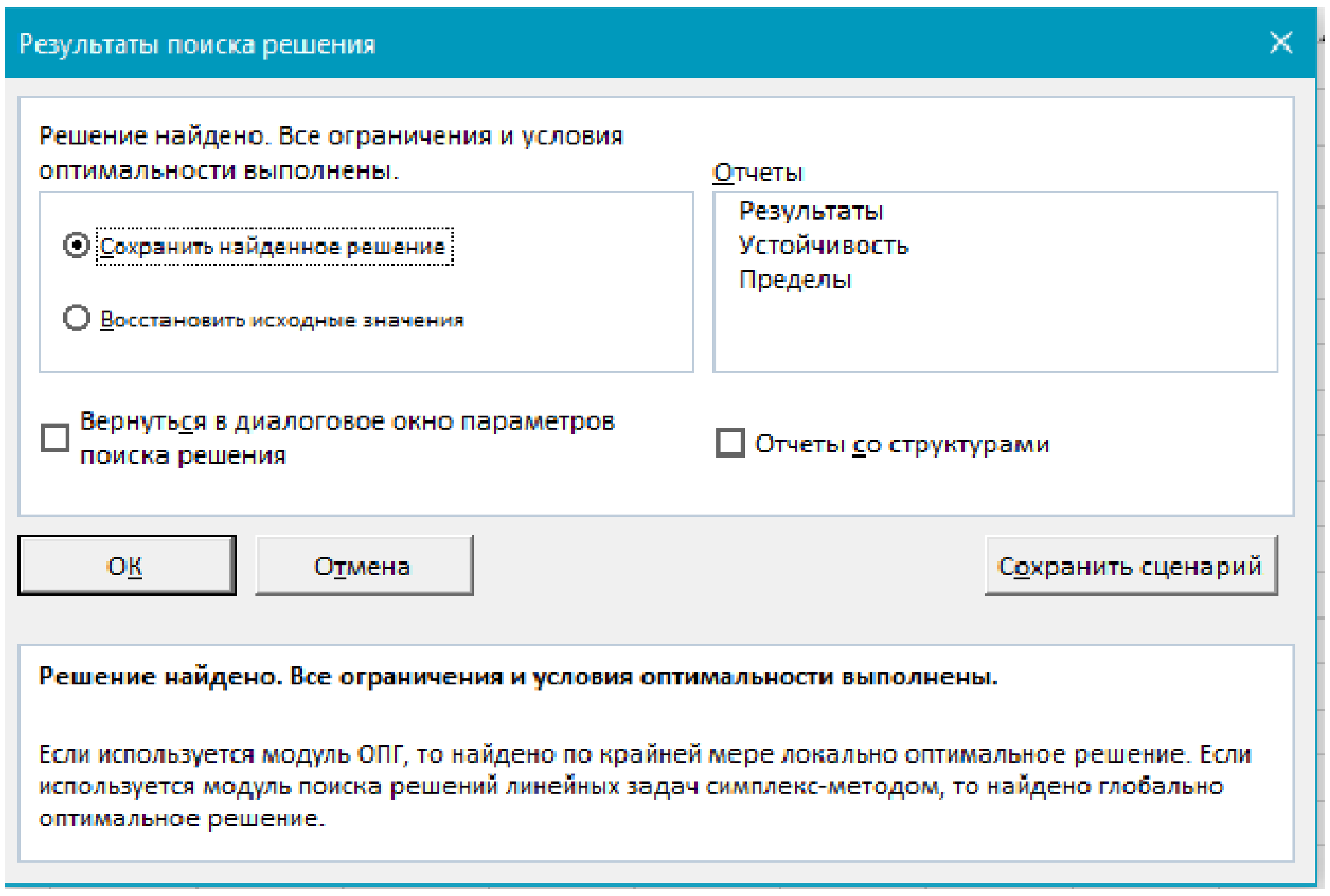


Рисунок 6.3 – Окно диалога **Результаты поиска решения**

Для просмотра результатов решения задачи необходимо нажать Enter или ОК. Получим максимальное значение функции равно 265,2173913, при значениях переменных $x_1 = 3{,}913043478$, $x_2 = 1{,}739130435$.

Понятие двойственности дает ощутимые практические результаты при построении алгоритмов решения задач линейного программирования. Для каждой задачи линейного программирования можно построить другую задачу, называемую двойственной.

Запишем обе задачи:

Прямая

$$F(x) = \sum_{j=1}^{n} c_j x_j (\min)$$

при ограничениях:

$$\sum_{j=1}^{n} a_{ij} x_j \ge b_i, i \in M¿, \quad i = (\overline{1,k})$$

$$\sum_{j=1}^{n} a_{ij} x_j = b_i, \quad i \in M¿, \quad i = (\overline{k+1,m})$$

$$x_j \ge 0, \quad j = (\overline{1,l},)$$

Двойственная

$$\phi(y) = \sum_{i=1}^{m} b_i y_i (\max)$$

при ограничениях:

$$\sum_{i=1}^{m} a_{ji} y_i \le c_j, j \in J, \quad j = (\overline{1,l})$$

$$\sum_{i=1}^{m} a_{ji} y_i = c_j, \quad j = (\overline{l+1,n})$$

$$y_i \ge 0, \quad i = (i = \overline{1,k}).$$

Симметричность обеих задач очевидна.

Неравенству первой задачи соответствует не отрицательность переменной во второй. Равенству одной задачи соответствует свободная переменная другой, задача двойственная к двойственной есть исходная (прямая) задача.

Таким образом:

Во взаимно-двойственных задачах всегда:

1. Одна из задач является задачей (*max*), а другая – задачей (*min*), в системе ограничений задачи (*max*) неравенств записаны со знаком ≤, а в

системе ограничений задачи (*min*) – со знаком ≥.

2. Каждому ограничению одной задачи соответствует переменная другой задачи: номер переменной совпадает с номером ограничения, при этом ограничению, записанному в виде неравенства, соответствует переменная, связанная условием неотрицательности.

3. Коэффициенты целевой функции одной задачи соответственно равны свободным членам системы ограничений другой задачи.

4. Матрица условий одной задачи получается из матрицы условий другой задачи с помощью транспонирования.

Взаимно-двойственными являются следующие задачи:

Прямая

$$F(x)=\sum_{j=1}^{n} c_j x_j \quad (\min)$$

при ограничениях:

$$\sum_{j=1}^{n} a_{ij} x_j \ge b_i, \quad j=\overline{1,n}$$

$$x_j \ge 0,\ j=\overline{1,n}.$$

Двойственная

$$\varphi(y)=\sum_{i=1}^{m} b_i y_i \quad (\max)$$

при ограничениях:

$$\sum_{i=1}^{m} a_{ji} y_i \le c_j, \quad i=\overline{1,m}$$

$$y_i \ge 0, i=\overline{1,m}.$$

Канонический вид

Прямая

$$F(x)=\sum_{j=1}^{n} c_j x_j \quad (min)$$

при ограничениях:

$$\sum a_{ij} x_j - b_i, i=\overline{1,m}$$

$$x_j \ge 0,\ j=\overline{1,n}.$$

Двойственная

$$\varphi(y)=\sum_{i=1}^{m} b_i y_i \quad (max)$$

при ограничениях:

$$\sum_{i=1}^{m} a_{ji} y_i = c_j, \quad j=\overline{1,n}$$

$$y_i \ge 0, i=\overline{1,m}.$$

Пример 2. Рассмотрим пару двойственных задач.

Прямая задача. Найти:

$$F=5x_1-x_2+8x_3-x_4 (\max)$$

при ограничениях:

$$\begin{cases} 2x_1+5x_2-x_3+7x_4=2 \\ x_1-x_2+5x_3-x_4\le 3 \end{cases}$$

$$x_1 \ge 0, x_3 \ge 0$$

Двойственная задача. Найти:

$$\Phi=2y_1+3y_2+5y_3(\min)$$

при ограничениях:

$$\begin{cases} 2y_1+y_2+y_3\ge 5 \\ 5y_1-y_2-y_3=-1 \\ -y_1+5y_2+3y_3\ge 8 \end{cases}$$

$$y_2 \ge 0$$

Запишем матрицы соответствующих задач: $A=\begin{vmatrix} 2 & 5 & -1 & 7 \\ 1 & -1 & 5 & -1 \\ 1 & -1 & 3 & 7 \end{vmatrix}$,

$A^T=\begin{vmatrix} 2 & 1 & 1 \\ 5 & -1 & -1 \\ -1 & 5 & 3 \\ 7 & -1 & 7 \end{vmatrix}$. $C = \begin{vmatrix} 5 & -1 & 8 \end{vmatrix}$; $C^T=\begin{vmatrix} 5 \\ -1 \\ 8 \end{vmatrix}$; $B=\begin{vmatrix} 2 \\ 3 \\ 5 \end{vmatrix}$; $B^T=\begin{vmatrix} 2 & 3 & 5 & 3 \end{vmatrix}$;

$A^T=\begin{vmatrix} 2 & -1 & 1 & 2 \\ 2 & 1 & 1 & 1 \\ -1 & 4 & -2 & -2 \end{vmatrix}$.

Наиболее часто встречаются следующие частные случаи взаимно-двойственных задач:

1) если $I = 0$, а $J=N=\{1,2,\dots,n\}$, то имеем симметричную пару;

2) если $I=M=\{1,2,\dots,m\}$, а $J=N=\{1,2,\dots,n\}$, то имеем не симметричную пару.

Рассмотрим симметричную пару двойственных задач.

Прямая задача. Найти:

$$F(x)=\sum_{j=1}^{n} c_j x_j \quad \text{(max)}$$

при ограничениях:

$$\sum_{j=1}^{n} a_{ij} x_j \le b_i, \quad i=\overline{1,m}$$

$$x_j \ge 0,\ j=\overline{1,n}.$$

Двойственная задача. Найти:

$$\phi(y)=\sum_{i=1}^{m} b_i y_i \quad \text{(min)}$$

при ограничениях:

$$\sum_{i=1}^{m} a_{ji} y_i \ge c_j, \quad j=\overline{1,n}$$

$$y_i \ge 0,\ i=\overline{1,m}.$$

Рассмотрим несимметричные двойственные задачи.

В несимметричных двойственных задачах система ограничений исходной задачи задается в виде неравенств, причем в последней переменные могут быть отрицательными.

Прямая задача. Найти:

$$F(x)=\sum_{j=1}^{n} c_j x_j \quad \text{(max)}$$

при ограничениях:

$$\sum_{j=1}^{n} a_{ij} x_j = b_i, \quad i=\overline{1,m}$$

Двойственная задача. Найти:

$$\varphi(y)=\sum_{i=1}^{m} b_i y_i \quad \text{(min)}$$

при ограничениях:

$$\sum_{i=1}^{m} a_{ji} y_i \ge c_j, \quad j=\overline{1,n}$$

$x_j \geq 0,\ j=\overline{1,n}.$ $\qquad\qquad$ $y_i \geq 0, i=\overline{1,m}.$

Простейшие свойства взаимно однозначных двойственных задач:

- если x допустимое решение прямой задачи, а y допустимое решение двойственной задачи, то $F(x) \leq \phi(y)$;
- если x и y допустимые решения прямой задачи и двойственной задачи, и $F(x) = \varphi(y)$, то x и y – оптимальные решения этих задач.

Свойства двойственных задач линейного программирования

Для взаимно-двойственных задач имеет место один из взаимоисключающих случаев, т.е. справедливо одно и только одно из следующих утверждений:

1. В прямой и двойственной задачах имеются оптимальные решения, причём значения целевых функций на оптимальных решениях совпадают, т.е. $\min f(x) = \max \phi(y)$.
2. В прямой задаче допустимо множество не пустое, а целевая функция на этом множестве не ограничена сверху. При этом у двойственной задачи будет пустое допустимое множество.
3. В двойственной задаче допустимо множество не пусто, а целевая функция на этом множестве не ограничена снизу. При этом у прямой задачи допустимое множество оказывается пустым.
4. Обе из рассматриваемых задач имеют пустые допустимые множества.

Первая (теорема двойственности):

Если из пары двойственных задач одна обладает оптимальным планом, то и другая имеет решение, причем для экстремальных решений линейных функций выполняется соотношение $\min F(x) = \max \phi(y)$, если линейная функция одной из задач не ограничена, то и другая не имеет решений.

Вторая теорема двойственности

Пусть $X=(x_1,\ x_2,\ \ldots,x_j,\ \ldots,x_n)$ – допустимое решение прямой задачи, а $Y=\{y_1,\ y_2,\ \ldots,\ y_i,\ldots,y_m\}$ – допустимое решение двойственной задачи. Для того, чтобы допустимые решения X и Y прямой двойственной задачи были оптимальными необходимо и достаточно, чтобы выполнялись следующие соотношения:

а) $Y(A\cdot X\text{-}b) = 0$ иначе

$$y_i\left(\sum_{j=1}^{n} a_{ij}\cdot x_j - b_i\right)=0$$

$$x_j\left(-\sum_{i=1}^{m} a_{ji}\cdot y_y + c_j\right)=0$$

б) $(C\text{-}Y\cdot A)X = 0$

Условия:

а) равносильно следующему:

если $y_i > 0$, то $\sum_{j=1}^{n} a_{ij} \cdot x_j = b_i, i = 1, \ldots, m$;

если $y_i = 0$, то $\sum_{j=1}^{n} a_{ij} \cdot x_j > b_i;$

б) равносильно следующему:

если $x_j > 0$, то $c_j = \sum_{i=1}^{m} a_{ji} \cdot y_i, j = 1, \ldots, n$;

если $x_j = 0$, то $c_j > \sum_{i=1}^{m} a_{ji} \cdot y_i$.

Может случиться, что одновременно $y_i = 0$ и $\sum_{j=1}^{n} a_{ij} \cdot x_j = b_i$. Но всегда существует по крайней мере одна пара.

Запишем двойственные задачи в векторном виде.

Несимметричные двойственные задачи

Прямая. Найти матрицу столбец $X = |x_1 \quad x_2 \quad \ldots \quad x_i \quad \ldots \quad x_n|$, которая удовлетворяет ограничениям $A \cdot X = B$, x ≥ 0 и минимизирует функцию $F = C \cdot X$.

Двойственная. Найти матрицу строку $Y = |y_1 \quad y_2 \quad \ldots \quad y_i \quad \ldots \quad y_m|$, которая удовлетворяет ограничениям $Y \cdot A \leq C$ и максимизирует линейную функцию $\varphi = Y \cdot B$.

В общих задачах: $C = |c_1 \quad c_2 \quad \ldots \quad c_j \quad \ldots \quad c_n|$ – матрица строка, представляющая собой коэффициенты при неизвестных в целевой функции; $B = |b_1 \quad b_2 \quad \ldots \quad b_i \quad \ldots \quad b_m|$ – матрица столбец свободных членов в системе ограничений; $A = |a_{ij}|$ – матрица коэффициентов при неизвестных в системе ограничений.

***Пример* 3.** Найти $F = x_2 - x_4 - 3x_5 (\min)$ при ограничениях:

$$\begin{cases} x_1 + 2x_2 - x_4 + x_5 = 1 \\ -4x_2 + x_3 + 2x_4 - x_5 = 2 \end{cases}$$

Матрица строка $C = |0 \quad 1 \quad 0 \quad -1 \quad -3 \quad 0|$ – коэффициенты при неизвестных в целевой функции. Матрица столбец $B = \begin{vmatrix} 1 \\ 2 \\ 5 \end{vmatrix}$ – свободные члены в системе ограничений, $A = \begin{vmatrix} 1 & 2 & 0 & -1 & 1 & 0 \\ 0 & -4 & 1 & 2 & -1 & 0 \\ 0 & 3 & 0 & 0 & 1 & 1 \end{vmatrix}$ – матрица коэффициентов при неизвестных в системе ограничений.

Пусть $X = |x_1 \quad x_2 \quad \dots \quad x_i \quad \dots \quad x_n|$ – допустимое решение прямой задачи. Вектор X является оптимальным решением этой задачи тогда и только тогда, когда среди решений системы уравнений $\sum_{i=1}^{m} a_{ji} \cdot y_i - c_j = 0$, при $x_j \neq 0$, $y_i = 0$, при $\sum_{j=1}^{n} a_{ij} \cdot x_j < b_i$ содержится хотя бы одно допустимое решение двойственной задачи.

***Пример* 4.** Вектор X = (3; 0,1; –3) – допустимое решение задачи $F = -2x_1 - x_2 + x_3 + x_4 (\max)$

при ограничениях:

$$\begin{cases} x_1 - x_2 + 2x_3 = 2 \\ 3x_2 - 7x_3 + 3x_4 = 2 \end{cases}$$

В данном случае соотношений $\sum_{j=1}^{n} a_{ij} \cdot x_j < b_i$.

Симметричные двойственные задачи

Прямая (исходная):

Найти матрицу-столбец $X = |x_1 \quad x_2 \quad \dots \quad x_i \quad \dots \quad x_n|$, которая удовлетворяет системе ограничений: $A \cdot X \geq B$; $X \geq 0$ и минимизирует функцию $F = C \cdot X$.

Двойственная задача:

Найти матрицу-строку $Y = |y_1 \quad y_2 \quad \dots \quad y_i \quad \dots \quad y_m|$, которая удовлетворяет системе ограничений: $Y \cdot A \leq C$; $Y \geq 0$ и максимизирует линейную функцию $\varphi = Y \cdot B$.

***Пример* 5.** Найти:

$$F = x_1 + 2x_2 + 3x_3 \quad (\min)$$

при ограничениях:

$$2x_1+2x_2-x_3\geq 2;\quad x_1-x_2-4x_3\leq -3;\quad x_1+x_2-2x_3\geq 6$$

x_j>0 (j = 1, 2, 3).

Чтобы привести систему ограничений к необходимому виду ($A\cdot X\geq B$), необходимо второе неравенство умножить на (–1), после чего запишем соответствующие матрицы:

$C = \begin{pmatrix}1 & 2 & 3\end{pmatrix}$, $B = \begin{pmatrix}2\\3\\6\\3\end{pmatrix}$; $A = \begin{pmatrix}2 & 2 & -1\\-1 & 1 & 4\\1 & 1 & -2\\2 & 1 & -2\end{pmatrix}$; $C^T = \begin{pmatrix}1\\2\\3\end{pmatrix}$; $B^T = \begin{pmatrix}2 & 3 & 5 & 3\end{pmatrix}$;

$A^T = \begin{pmatrix}2 & -1 & 1 & 2\\2 & 1 & 1 & 1\\-1 & 4 & -2 & -2\end{pmatrix}$.

В результате двойственная задача имеет вид:

Найти:

$$\varphi = 2y_1+3y_2+6y_3+3y_4\ (\max)$$

при ограничениях:

$$2y_1-y_2+y_3+2y_4\leq 1;\quad 2y_1+y_2+y_3+y_4\leq 2$$

$y_j\geq 0$, (j=1, 2, 3, 4).

Аппаратура и материалы. Для выполнения лабораторной работы необходимо использовать следующее: аппаратное обеспечение: персональный компьютер и программное обеспечение: операционную систему Windows 10 и выше; Microsoft Office13 и выше, системы компьютерной математики Machcad и MATLAB R2017a и выше.

Указания по технике безопасности. Самостоятельно не производить: установку и удаление программного обеспечения; ремонт персонального компьютера. Соблюдать правила технической эксплуатации и техники безопасности при работе с электрооборудованием.

Методика и порядок выполнения работы

Выполните предложенные задания, предварительно решив все приведенные в данной работе примеры.

Задание 6.1. Постановка задачи линейного программирования.

Построить математическую модель планирования производства. Для изготовления трех видов продукции P_1, P_2, P_3 используют три вида сырья S_1, S_2, S_3. Запасы сырья, число единиц сырья, затрачиваемых на изготовление единицы продукции, прибыль, получаемая от единицы продукции P_1, P_2, P_3 – соответственно приведены в таблице 6.1 (цифры условные).

Таблица 6.1 – Запасы сырья S_1, S_2, S_3, число единиц сырья, затрачиваемых на изготовление единицы продукции, прибыль, получаемая от единицы продукции P_1, P_2, P_3.

Вид сырья	Число единиц сырья, затрачиваемых на изготовление единицы продукции			Запасы сырья
	P_1	P_2	P_3	
S_1	4	2	1	150 000
S_2	6	0	2	170 000
S_3	0	2	4	100 000
Прибыль от ед. прод.	100	150	200	

Необходимо составить такой план производства продукции, при котором прибыль от ее реализации будет максимальной.

- Записать задачу в каноническом виде и решить её в Excel, используя инструмент **Поиск решения**.

Задание 6.2. Индивидуальное. Для исходной задачи записать двойственную задачу, решить обе задачи с помощью инструментального средства **Поиск решения** и проанализировать найденные результаты, используя теоремы двойственности.

1. $f=4x_1+5x_2+6x_3 \to$ (min)

при ограничениях:

$$\begin{cases} x_1+x_2+x_3\ge 5 \\ x_1-x_2+2x_3\ge 1 \\ x_1-x_2-4x_3\le -3 \\ x_1-x_2+8x_3\ge 4 \end{cases}$$

$$x_j\ge 0,\ \left(j=\overline{1,3}\right).$$

2. $f=2x_1-2x_2+3x_3+4x_4$ (max)

при ограничениях:

$$\begin{cases} x_1+2x_2+x_3+x_4\le 2 \\ 2x_1-x_2+2x_3-3x_4\ge 3 \\ 3x_1+4x_2-5x_3+2x_4\le 4 \end{cases}$$

$$x_j\ge 0,\ \left(j=\overline{1,4}\right).$$

3. $f=5x_1-x_2-4x_3 \to \left(\max\right)$

при ограничениях:

$$\begin{cases} -x_2+2x_3\ge 9 \\ -x_1+x_2\ge 1 \\ x_1+x_2-3x_3\ge 8 \\ x_1-x_3\le 4 \end{cases}$$

$$x_j\ge 0,\ \left(j=\overline{1,3}\right).$$

4. $f=2x_1+3x_3+\frac{5}{2}x_3 \to \left(\min\right)$

при ограничениях:

$$\begin{cases} 2x_1+x_2+3x_3\ge 6 \\ 2x_1+4x_2+3x_3\ge 16 \\ 3x_1+4x_2+2x_3\ge 12 \end{cases}$$

$$x_j\ge 0,\ \left(j=\overline{1,3}\right).$$

5. $f=4x_1+6x_2+3x_3$ (min)

при ограничениях:

$$\begin{cases} 3x_1+x_2+2x_3\geq 9 \\ x_1+2x_2+3x_3\geq 8 \\ x_1+6x_2\geq 12 \end{cases}$$

$$x_j\geq 0, \diagup j=\left(\overline{1,3}\right).$$

6. $f=2x_1+3x_2+7x_3$ (max)

при ограничениях:

$$\begin{cases} 2x_1+3x_2+x_3\leq 9 \\ -x_1-2x_2-3x_3\geq -8 \\ 5x_1+6x_2\leq 12 \end{cases}$$

$$x_j\geq 0, \diagup j=\left(\overline{1,3}\right).$$

7. $f=6x_1+8x_2+x_3 \diagup (\max)$

при ограничениях:

$$\begin{cases} x_1+x_2+x_3\leq 3 \\ x_1+2x_2\leq 4 \end{cases}$$

$$x_j\geq 0, \diagup j=\left(\overline{1,3}\right).$$

8.
$f=27x_1+10x_2+15x_3+28x_4 \diagup (\max)$

при ограничениях:

$$\begin{cases} 3x_1+2x_2+x_3+2x_4\leq 2 \\ 3x_1+x_2+3x_3+4x_4\leq 5 \end{cases}$$

$$x_j\geq 0, \diagup j=\left(\overline{1,4}\right).$$

9.
$f=3x_1+2x_2-6x_3 \diagup (\max)$

при ограничениях:

$$\begin{cases} 2x_1-3x_2+x_3\leq 18 \\ -3x_1+2x_2-2x_3\leq 24 \\ x_1+3x_2-4x_3\leq 36 \end{cases}$$

$$x_j\geq 0, \diagup j=\left(\overline{1,3}\right).$$

10. $f=3x_1-7x_2-4x_3 \diagup (\max)$

при ограничениях:

$$\begin{cases} -2x_1-3x_2-2x_3\leq 12 \\ -4x_1-4x_2-3x_3\leq 24 \\ 5x_1+5x_2+3x_3\leq 15 \end{cases}$$

$$x_j\geq 0, \diagup j=\left(\overline{1,3}\right).$$

11.
$f=x_1+3x_2-5x_4 \diagup (\max)$

при ограничениях:

$$\begin{cases} 2x_1+4x_2+x_3+2x_4=28 \\ -3x_1+5x_2-3x_4\leq 30 \\ 4x_1-2x_2+?8x_4\leq 32 \end{cases}$$

$$x_j\geq 0, \diagup j=\left(\overline{1,4}\right).$$

12 $f=3x_1+2x_5-5x_6 \diagup (\max)$

при ограничениях:

$$\begin{cases} 2x_1+x_2-3x_5+5x_6=30 \\ 4x_1+x_3+2x_5-4x_6=28 \\ -3x_1+x_4-3x_5+6x_6=24 \end{cases}$$

$$x_j\geq 0, \diagup j=\left(\overline{1,6}\right).$$

13. $f=x_1+x_2-3x_3 \diagup (\max)$

при ограничениях:

$$\begin{cases} -x_2-2x_3\leq -7 \\ -x_1+3x_2\geq 10 \\ x_1+x_2-2x_3\geq 4 \\ x_1+x_2-x_3\leq 4 \end{cases}$$

$$x_j\geq 0, \diagup \left(j=\overline{1,3}\right).$$

14. $f=2x_1-3x_3+4x_3 \diagup (\min)$

при ограничениях:

$$\begin{cases} x_1+2x_2+3x_3\geq 5 \\ 2x_1+3x_2-4x_3\geq 10 \\ 3x_1+x_2+-7x_3\geq 13 \end{cases}$$

$$x_j\geq 0, \diagup \left(j=\overline{1,3}\right).$$

15. $f=4x_1-5x_2+2x_3 \diagup$
(min)

16. $f=x_1+4x_2-7x_3$ (min)

при ограничениях:

при ограничениях:

$$\begin{cases} 5x_1-3x_2+4x_3\geq 9 \\ x_1-4x_2+3x_3\geq 8 \\ x_1+x_2\geq 10 \end{cases}$$

$$x_j\geq 0,\; j=\left(\overline{1,3}\right).$$

17. $f=3x_1-7x_2+4x_3\;(\min)$

при ограничениях:

$$\begin{cases} x_1-8x_2+3x_3\geq 5 \\ 3x_1+4x_2-7x_3\geq 18 \\ x_1+7x_2\geq 12 \end{cases}$$

$$x_j\geq 0,\; j=\left(\overline{1,3}\right).$$

19. $f=-4x_1+x_2-3x_3$ (max)

при ограничениях:

$$\begin{cases} 6x_1+3x_2-4x_3\geq 15 \\ 2x_1+4x_2-3x_3\geq 18 \\ x_1+7x_2\geq 1 \end{cases}$$

$$x_j\geq 0,\; j=\left(\overline{1,3}\right).$$

$$\begin{cases} 4x_1-5x_2+x_3\leq 3 \\ -x_1-2x_2-3x_3\geq -7 \\ 5x_1-x_2\leq 10 \end{cases}$$

$$x_j\geq 0,\; j=\left(\overline{1,3}\right).$$

18. $f=-6x_1+2x_2-9x_3$ (max)

при ограничениях:

$$\begin{cases} 7x_1-5x_2+3x_3\leq 13 \\ -9x_1+3x_2-5x_3\geq -8 \\ x_1-4x_2\leq 13 \end{cases}$$

$$x_j\geq 0,\; j=\left(\overline{1,3}\right).$$

20. $f=11x_1+4x_2-7x_3$ (min)

при ограничениях:

$$\begin{cases} 2x_1-3x_2+9x_3\leq 3 \\ -4x_1-9x_2-6x_3\geq 7 \\ 5x_1-8x_2\leq 1 \end{cases}$$

$$x_j\geq 0,\; j=\left(\overline{1,3}\right).$$

Задание 4. Для двойственной симметричной задачи записать исходную и найти ее решение:

$$j=2y_1+4x_2+12y_4\;(\min)$$

при ограничениях:

$$\begin{cases} y_1+2y_2+y_3+4y_4\geq 10 \\ 2y_1+y_2-2y_3+3y_4\geq 4 \end{cases}$$

$$y_j\geq 0,\;\left(j=\overline{1,4}\right).$$

Содержание отчета и его форма

Подготовьте отчет, в котором опишите технологию решения задач линейного программирования средствами Excel, используя задания своего варианта. Отчет по лабораторной работе должен содержать:

1) название работы;
2) цель лабораторной работы;
3) условия выполненных заданий и их решение;
4) технологию решения задач с помощью прикладного программного обеспечения, выводы;
5) ответы на контрольные вопросы.

Вопросы для защиты работы

1. Дайте определение математического программирования.

2. Что называется математической моделью оптимизационной задачи?
3. Как строится математическая модель?
4. Как перейти от неравенств к уравнениям?
5. Какие переменные называются дополнительными (слабыми)?
6. Расскажите известные вам методы непосредственного решения простейших оптимизационных задач.
7. В чем заключается сущность двойственности в линейном программировании?
8. Сформулируйте 1-ю и 2-ю теорему двойственности.
9. Какие задачи линейного программирования относятся к симметричным и несимметричным? В чем их отличие?
10. Как по решению исходной задачи найти решение двойственной и наоборот?

ЛАБОРАТОРНАЯ РАБОТА 7
Построение и анализ моделей задач транспортного типа, решение задач методом потенциалов

Цель и содержание: Научиться строить и анализировать модели транспортной задачи.

Организационная форма занятий: решение проблемных задач, разбор конкретных ситуаций

Вопросы для обсуждения на лабораторном занятии: построение, решение и анализ модели транспортной задачи.

Формируемые компетенции: ОПК-7.1 ОПК-7.2 ОПК-8.1 ОПК-8.2

Теоретическое обоснование

Изучите теоретический материал по данной теме, используя материал изложенный ниже.

Классические транспортные задачи относятся к области линейного программирования, и математический аппарат решения этих задач хорошо отработан и известен. Сфера применения моделей задач транспортного типа обширна. Сюда можно отнести проблемы, например, связанные с планированием производства и перевозок, с созданием вычислительных и информационных систем, с распределениями ресурсов, запасов и так далее, то есть в формальных терминах транспортной задачи формируется большое число задач, отнюдь не связанных с перевозками продуктов.

Сформулируем математическую модель транспортной задачи.

Некоторый однородный продукт, сосредоточенный у m поставщиков A в количестве a_i (i = 1, 2, ..., m) единиц соответственно, необходимо доставить n потребителям B в количестве b_j (j = 1, 2, ..., n) единиц. Известна стоимость C_{ij} перевозки единицы груза от i-того поставщика к j-тому потребителю. Необходимо составить план перевозок, позволяющий вывезти все грузы, полностью удовлетворить потребности и имеющий минимальную стоимость.

Обозначим через x_{ij} количество единиц груза, запланированных к перевозке от i-того поставщика к j-тому потребителю; тогда условие задачи можно записать в виде таблицы 4.1, которую в дальнейшем называют *матрицей планирования.*

Таблица 7.1 – Матрица планирования

Поставщики A_i	Потребители, B_j				Запасы a_i
	B_1	B_2	…	B_n	
A_1	c_{11} x_{11}	c_{12} x_{12}	…	c_{1n} x_{1n}	a_1
A_2	c_{21} x_{21}	c_{22} x_{22}	…	c_{2n} x_{2n}	a_2
…	…	…	…	…	…
A_m	c_{m1} x_{m1}	c_{m2} x_{m2}	…	c_{mn} x_{mn}	a_m

Потребности, b_j	b_1	b_2	…	b_n	$\Sigma a_i=\Sigma b_j$

Составим математическую модель задачи. Так как от i-го поставщика к j-тому потребителю запланировано к перевозке x_{ij} единиц груза, то стоимость перевозки составит $c_{ij}x_{ij}$. Стоимость всего плана выразится двойной суммой:

$$Z=\sum_{i=1}^{m}\sum_{j=1}^{n} c_{ij}x_{ij}.$$

Систему ограничений получим из следующих условий задачи:

а) все грузы должны быть вывезены, то есть,

$$\sum_{j=1}^{n} x_{ij}=a_i \quad , (i = 1, 2, ..., m),$$

эти уравнения получаются из строк таблицы 7.1;

б) все потребности должны быть удовлетворены, то есть,

$$\sum_{i=1}^{m} x_{ij}=b_j \quad , (j = 1,2, ..., n),$$

уравнения получаются из столбцов таблицы 4.1.

Таким образом, математическая модель транспортной задачи имеет следующий вид. Найти наименьшее значение функции:

$$Z=\sum_{i=1}^{m}\sum_{j=1}^{n} c_{ij}x_{ij} \tag{7.1}$$

при ограничениях:

$$\left\{\sum_{i=1}^{m} x_{ij}=b_j, j=\overline{1,n} \text{¿¿¿¿}_{\text{¿}}\right. \text{¿} \tag{7.2}$$

В рассмотренной модели предполагается, что суммарные запасы равны суммарным потребностям, то есть

$$\sum_{i=1}^{m} a_i=\sum_{j=1}^{n} b_j \quad .$$

Такая модель называется *закрытой*.

Рассмотрим пример построения математической модели транспортной задачи, методы построения опорного плана и метод решения задачи.

Пример. Четыре предприятия могут производить некоторую однородную продукцию в количестве соответственно 100, 250, 200 и 300. На эту продукцию есть заказ от пяти потребителей соответственно в количестве 200, 200, 100, 100 и 250. Затраты с производством и доставкой единицы продукции задаются матрицей: $C = \begin{vmatrix} 10 & 7 & 4 & 1 & 4 \\ 2 & 7 & 10 & 6 & 11 \\ 8 & 5 & 3 & 2 & 2 \\ 11 & 8 & 12 & 16 & 13 \end{vmatrix}$.

Необходимо: построить математическую модель заданной транспортной задачи; найти опорный план методами: северо-западного угла; минимальной стоимости; двойного предпочтения; построить систему потенциалов и проверить найденные планы на оптимальность; найти оптимальный план методом потенциалов, взяв за первоначальный любой из опорных планов, не являющийся оптимальным.

Решение. *Построение математической модели заданной экономической системы.* Сформулируем математическую задачу: пусть некоторый однородный продукт, сосредоточен у $m = 4$ поставщиков A_i в количестве a_i ($i = 1, 2, ..., m$) единиц соответственно, необходимо доставить этот продукт n потребителям B_j в количестве b_j ($j = 1, 2, ..., n$) единиц. Известна стоимость c_{ij} перевозки единицы продукта от i-го поставщика к j-тому потребителю заданная матрицей стоимостей. Необходимо составить такой план перевозки продукта, который имеет минимальную стоимость.

Прежде чем составлять математическую модель заданной системы необходимо проверить условие закрытости задачи: а именно, что суммарные запасы равны суммарным потребностям, т. е. все запасы должны быть вывезены, а потребности удовлетворены:

$$\sum_{i=1}^{m=4} a_i = \sum_{j=1}^{n=5} b_j$$

$$\sum_{i=1}^{m=4} a_i = 100 + 250 + 200 + 300 = 850$$

$$\sum_{j=1}^{n=5} b_j = 200 + 200 + 100 + 100 + 250 = 850.$$

Так как суммарные потребности равны суммарным запасам, имеем закрытую транспортную задачу. Обозначим через x_{ij} количество единиц груза, запланированных к перевозке от i-го поставщика к j-тому потребителю, и запишем условие задачи в виде матрицы планирования (таблица 7.2).

Таблица 7.2 – Матрица планирования для задачи, представленной в примере

Поставщики, A_i	Потребители, B_j					Запасы a_i
	B_1	B_2	B_3	B_4	B_5	
A_1	10 x_{11}	7 x_{12}	4 x_{13}	1 x_{14}	4 x_{15}	100
A_2	2	7	10	6	11	250

	x_{21}	x_{22}	x_{23}	x_{24}	x_{25}	
A_3	8 x_{31}	5 x_{32}	3 x_{33}	2 x_{34}	2 x_{35}	200
A_4	11 x_{41}	8 x_{42}	12 x_{43}	16 x_{44}	13 x_{45}	300
Потребности b_j	200	200	100	100	250	$\sum_{i=1}^{m} a_i = \sum_{j=1}^{n} b_j$

Составим математическую модель задачи. Так как от i-го поставщика к j-тому потребителю запланировано к перевозке x_{ij} единиц груза, то стоимость перевозки составит $c_{ij}x_{ij}$. Стоимость всего плана выразится двойной суммой:

$$F = \sum_{i=1}^{m=4} \sum_{j=1}^{n=5} c_{ij} x_{ij}.$$

Систему ограничений получим из следующих условий задачи:

а) все продукты должны быть вывезены, то есть,

$$\sum_{j=1}^{n=5} x_{ij} = a_i, \quad (i = 1, 2, 3, 4),$$

эти уравнения получаются из строк таблицы 4.2;

б) все потребности должны быть удовлетворены, то есть,

$$\sum_{i=1}^{m=4} x_{ij} = b_j, \quad (j = 1,2, ..., 5),$$

уравнения получаются из столбцов таблицы 7.2.

Таким образом, математическая модель данной задачи имеет следующий вид:

$$F = \sum_{i=1}^{m=4} \sum_{j=1}^{n=5} c_{ij} x_{ij} (\min) \tag{7.3}$$

при ограничениях:

$$\left\{ \sum_{i=1}^{m=4} x_{ij} = b_j, j = \overline{1,5} ¿¿¿¿ \right. ¿ \tag{7.4}$$

$$c_{ij}=\begin{pmatrix} 10 & 7 & 4 & 1 & 4 \\ 2 & 7 & 10 & 6 & 11 \\ 8 & 5 & 3 & 2 & 2 \\ 11 & 8 & 12 & 16 & 13 \end{pmatrix}, \qquad a_i=\begin{pmatrix} 100 \\ 250 \\ 200 \\ 300 \end{pmatrix},$$

где

$$b_j=\begin{pmatrix} 200, & 200, & 100, & 100, & 250 \end{pmatrix}.$$

Иначе математическую модель можно записать в следующем виде:

$$F=10x_{11}+7x_{12}+4x_{13}+1x_{14}+4x_{15}+2x_{21}+7x_{22}+10x_{23}+6x_{24}+11x_{25}+8x_{31}+5x_{32}+3x_{33}+2x_{34}+2x_{35}+11x_{41}+8x_{42}+12x_{43}+16x_{44}+13x_{45}\to \min$$

при ограничениях:

Таким образом, математическая постановка данной задачи состоит в нахождении такого не отрицательного решения системы линейных уравнений, при котором целевая функция принимает минимальное значение.

Определение опорного плана методами: северо-западного угла; минимальной стоимости; двойного предпочтения. Найдем опорный план методом северо-западного угла. Для этого запишем условие задачи в виде таблицы 7.3. Не учитывая стоимости перевозки единицы груза, начинаем удовлетворение потребностей первого потребителя B_1 за счет запаса поставщика A_1.

Для этого сравниваем a_1=100 с b_1=200, $a_1<b_1$, меньший из объемов, т. е. =100ед. записываем в левый нижний угол клетки A_1B_1. Запасы первого поставщика полностью израсходованы, поэтому остальные клетки первой строки вычеркиваем. Потребности B_1 остались неудовлетворенными на 200 –100 = 100 ед. Сравниваем этот остаток с запасами поставщика A_2: т. к.

100 < 250, то 100 ед. записываем в клетку A_2B_1, чем полностью

удовлетворяем потребности потребителя B_1, оставшиеся клетки в первом столбце вычеркиваем.

У поставщика A_2 осталось 150 ед. груза. Удовлетворяем потребности потребителя B_2 за счет оставшихся у поставщика A_2 запасов. Для этого сравниваем этот остаток с потребностями потребителя B_2: 150<200, записываем 150 ед. в клетку A_2B_2 и, т. к. запасы A_2 полностью израсходованы, прочеркиваем остальные клетки второй строки. Потребности B_2 остались неудовлетворенными на 50ед. Удовлетворяем их за счет поставщика A_3 и переходим к удовлетворению B_3 за счет остатка, имеющегося у поставщика A_3, и т. д. Процесс продолжаем до тех пор, пока не удовлетворим всех потребителей за счет запасов поставщиков. На этом построение первоначального опорного план заканчивается.

Таким образом, в таблице 7.3 в правых верхних углах клеток стоят числа, определяющие стоимость перевозки единицы грузов, в левых нижних углах – числа, определяющие план перевозок. Проверим, является ли план, построенный в таблице опорным. Видим, что, начиная движение от занятой клетки A_1B_1, вернуться не только в нее, но и в любую другую занятую клетку, двигаясь только по занятым клеткам, невозможно. Следовательно, план является опорным. В то же время план является невырожденным, т. к. он содержит точно $m + n – 1$=4 + 5 – 1 = 8 занятых клеток.

Таблица 7.3 – Построение первоначального опорного плана методом северо-западного угла

Поставщики, A_i	Потребители, B_j					Запасы a_i
	B_1	B_2	B_3	B_4	B_5	
A_1	10 100	7 –	4 –	1 –	4 –	100
A_2	2 100	7 150	10 –	6 –	11	250
A_3	8 –	5 50	3 100	2 50	2	200
A_4	11 –	8 –	12 –	16 50	13 250	300
Потребности b_j	200	200	100	100	250	

Если же занятых клеток будет меньше, чем $m + n – 1$, то следует ввести в любую незанятую клетку объем груза равный нулю и, таким образом, вырожденный план привести к невырожденному.

Найдем общую стоимость составленного плана как сумму произведений объемов перевозок, стоящих в левом углу занятых клеток, на соответствующие стоимости в этих же клетках:

$F = 100·10+100·2+150·7+50·5+100·3+50·2+50·16+250·13=6950$(ед. стоимости).

При составлении первоначального опорного плана методом северо-западного угла стоимость перевозки единицы груза не учитывалась, поэтому построенный план далек от оптимального. Построение опорного плана удобно выполнять в табличном процессоре Excel. Для этого необходимо ввести данные таблицы 7.3 на рабочий лист Excel, затем: эту таблицу скопировать и удалить все числа, оставив название полей; ввести формулы для подсчета сумм перевозок по строкам и столбцам Затем в соответствующие ячейки по строке «Потребности» и столбцу «Запасы»; заполнить ячейки таблицы значениями перевозок.

На рисунке 7.4 представлен рабочий лист Excel с введенными в первой таблице исходными данными задачи: стоимостями, запасами и потребностями, а во второй таблице – формулами для подсчета сумм перевозок по строкам:=СУММ(K3:O3) и столбцам: =СУММ(K3:K6) и найденным опорным планом.

В ячейке A10 введена формула, с помощью которой определяется целевая функция: =СУММПРОИЗВ(B3:F6;K3:O6). Если при составлении опорного плана учитывать стоимость перевозки единицы груза, то план будет значительно ближе к оптимальному.

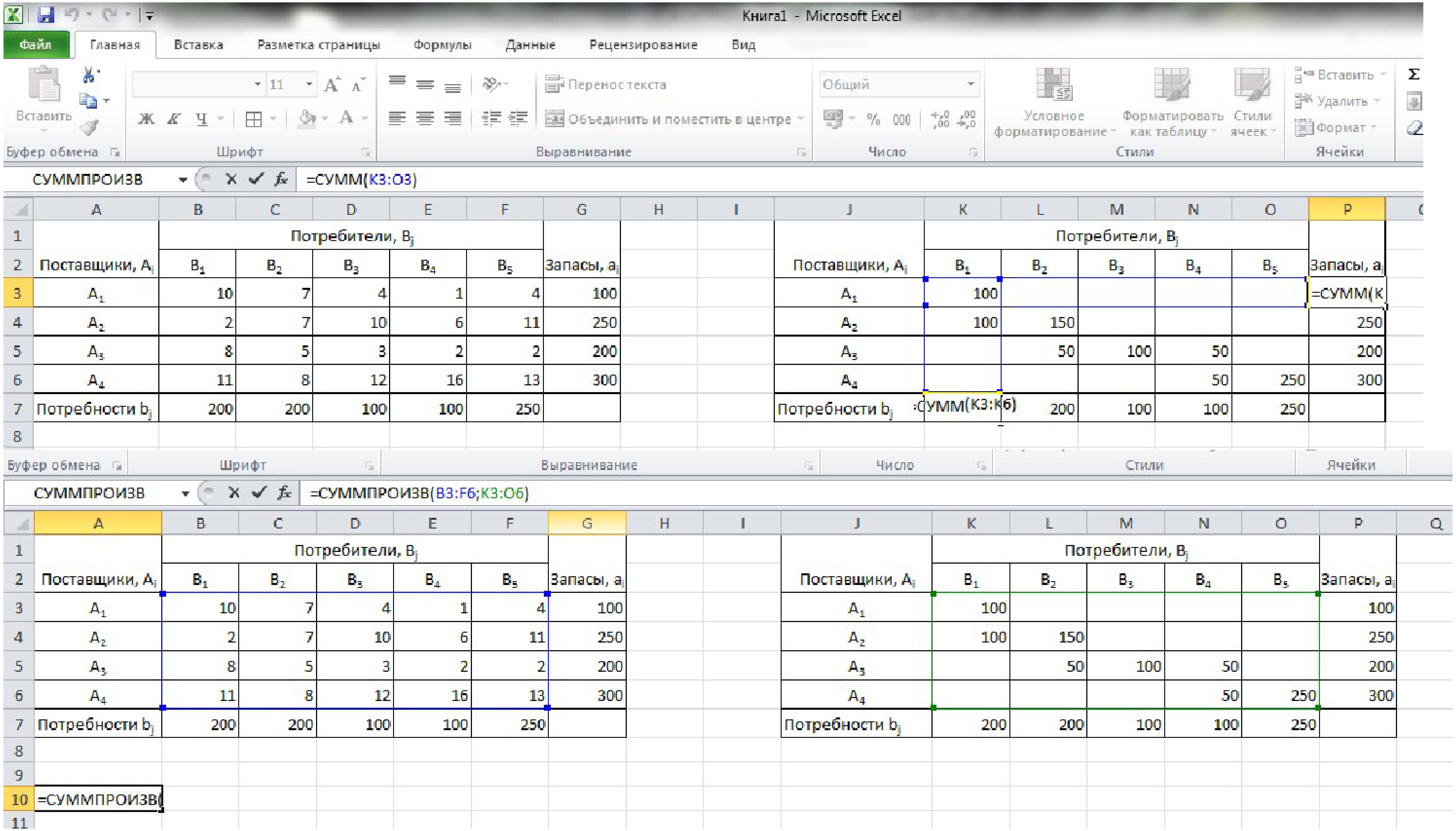


Рисунок 7.4 – Построение первоначального опорного плана методом северо-западного угла

Найдем опорный план методом минимальной стоимости. Сущность метода заключается в том, что из всей таблицы стоимостей выбирают наименьшую и в клетку, которая ей соответствует, помещают меньшее из чисел a_i или b_j. Затем из рассмотрения исключают либо строку, соответствующую поставщику, запасы которого полностью израсходованы, либо столбец, соответствующий потребителю, потребности которого полностью удовлетворены, либо и строку, и столбец, если израсходованы запасы поставщика и удовлетворены потребности потребителя. Из оставшейся части таблицы стоимостей снова выбирают наименьшую стоимость, и процесс распределения запасов продолжают, пока все запасы не будут распределены, потребности удовлетворены. Составим с помощью этого метода опорный план уже рассмотренной задачи и запишем ее условие в таблицу 7.5. Выбираем в таблице наименьшую стоимость (это стоимость, помещенная в клетке A_1B_4), так как $a_1=b_4$, 100 ед. груза помещаем в этой клетке и исключаем из рассмотрения первую строку и четвертый столбец. В оставшейся таблице стоимостей наименьшей является стоимость, расположенная в клетке A_2B_1 и в клетке A_3B_5. Заполняем любую из них, например, A_2B_1.

Имеем 200<250, следовательно, записываем в нее 200 и исключаем из рассмотрения столбец B_1. В клетку A_3B_5 записываем 200 ед. и исключаем из рассмотрения строку A_3.

Таблица 7.5 – Построение первоначального опорного плана методом минимальной стоимости

Поставщики, A_i	Потребители, B_j					Запасы
	B_1	B_2	B_3	B_4	B_5	a_i
A_1	10 –	7 –	4 –	1 100	4 –	100
A_2	2 200	7 50	10 –	6 –	11 –	250
A_3	8 –	5 –	3 –	2 –	2 200	200
A_4	11 –	8 150	12 100–	16 –	13 50	300
Потребности b_i	200	200	100	100	250	

В оставшейся таблице стоимостей снова выбираем наименьшую стоимость и продолжаем процесс до тех пор, пока все запасы не будут распределены, потребности удовлетворены. В результате получен план X=(x_{14}=100; x_{21}=200; x_{22}=50; x_{35}=200; x_{42}=150; x_{43}=100; x_{45}=50), остальные значения переменных равны нулю. План не содержит циклов и состоит из семи положительных перевозок, следовательно, является опорным планом. Определим его стоимость:

$$F = 100 \cdot 1+200 \cdot 2+50 \cdot 7+200 \cdot 2+ 150 \cdot 8+ 100 \cdot 12++50 \cdot 13=4300 \text{ (ед.)}.$$

Стоимость плана перевозок значительно меньше, следовательно, он ближе к оптимальному.

Определим опорный план методом двойного предпочтения. Если таблица стоимостей велика, то перебор всех элементов затруднителен. В этом случае используют метод двойного предпочтения, суть которого заключается в следующем:

В каждом столбце отмечают знаком «V» клетку с наименьшей стоимостью. Затем то же проделывают в каждой строке. В результате некоторые клетки имеют отметку «VV» (рисунок 7.6).

В них находится минимальная стоимость, как по столбцу, так и по строке. В эти клетки помещают максимально возможные объемы перевозок, каждый раз исключая из рассмотрения соответствующие столбцы или строки. Затем распределяют перевозки по клеткам, отмеченным знаком «V». В оставшейся части таблицы перевозки распределяют по наименьшей стоимости.

Таблица 7.6 – Выделенные клетки с минимальной стоимостью

Поставщики, A_i	Потребители, B_j					Запасы a_i
	B_1	B_2	B_3	B_4	B_5	
A_1	10	7	4	VV 1	4	100
A_2	VV 2	7	10	6	11	250
A_3	8	V 5	V 3	V 2	VV 2	200
A_4	11	V 8	12	16	13	300
Потребности b_j	200	200	100	100	250	

Затем сначала заполняем клетки A_1B_4, A_2B_1, A_3B_5, потом клетку A_3B_5. В оставшейся части таблицы последовательно заполняем клетки по минимальной стоимости A_3B_2, A_3B_3, A_3B_4 , A_4B_2, (таблица7.7).

Таблица 7.7 – Опорный план, найденный методом двойного предпочтения

Поставщики, A_i	Потребители, B_j					Запасы a_i
	B_1	B_2	B_3	B_4	B_5	
A_1	10 –	7 –	4 –	VV 1 100	4 –	100
A_2	VV 2 200	7 –	10 –	6 –	11 50	250
A_3	8 –	V 5 –	V 3 –	V 2 –	VV 2 200	200
A_4	11 –	V 8 200	12 100	16 –	13 –	300
Потребности b_j	200	200	100	100	250	

Найдем стоимость полученного опорного плана.

$F = 100 \cdot 1 + 200 \cdot 2 + 50 \cdot 10 + 200 \cdot 2 + 200 \cdot 8 + 50 \cdot 12 + 50 \cdot 13 = 4250$ (ед.).

Таким образом, наименьшую стоимость имеет опорный план, полученный методом двойного предпочтения, следовательно, он наиболее близок к оптимальному плану. Однако отсюда не следует вывод, что с помощью метода двойного предпочтения всегда получают лучший первоначальный план по сравнению с методом минимальной стоимости.

Можно показать, что задача линейного программирования двойственная классической транспортной задаче (7.1, 7.2) состоит в максимизации целевой функции:

$$\phi(\alpha_1, \ldots, \alpha_m, \beta_1, \ldots, \beta_n) = \sum_{i=1}^{m=4} a_i \alpha_i + \sum_{j=1}^{n=5} b_j \beta_j \qquad (7.5)$$

при ограничениях

$$\alpha_i + \beta_j \le c_{ij}, i = \overline{1,m}; j = \overline{1,n}, \qquad (7.6)$$

где числа $\alpha_i, i = \overline{1,m}$ и $\beta_j, j = \overline{1,n}$ называются соответственно потенциалами поставщиков и потребителей. Для того чтобы решение классической транспортной задачи было оптимальным необходимо выполнение следующей теоремы:

Теорема. Если план $X^* = (x^*_{ij})$ транспортной задачи является оптимальным, то ему соответствует система из $m+n$ чисел $\alpha_i, i = \overline{1,m}$ и $\beta_j, j = \overline{1,n}$, удовлетворяющих условиям:

$$\alpha_i + \beta_j = c_{ij}, \text{ для } x^*_{ij} > 0, \qquad (7.8)$$

$$\alpha_i + \beta_j \le c_{ij}, \text{ для } x^*_{ij} = 0. \qquad (7.9)$$

Из теоремы следует: для того, чтобы первоначальный опорный план был оптимальным, необходимо выполнение следующих условий:

- для каждой занятой клетки сумма потенциалов должна быть равна стоимости единицы перевозки, стоящей в этой клетке $\alpha_i + \beta_j = c_{ij}$;
- для каждой незанятой клетки сумма потенциалов должна быть меньше или равна стоимости единицы перевозки, стоящей в этой клетке $\alpha_i + \beta_j \le c_{ij}$.

Если хотя бы одна незанятая клетка не удовлетворяет этому условию, то опорный план не является оптимальным и его можно улучшить, перемещая в эту клетку некоторое количество груза.

Используя первое условие теоремы 7.6, построим систему потенциалов (таблица 7.6), используя таблицу 7.7, предварительно проверив, сколько занятых клеток имеет таблица. Если занятых клеток меньше, чем $m+n-1$ заполняем недостающие клетки нулевыми перевозками.

Систему потенциалов можно построить только для невырожденного плана, который, как известно, содержит $m+n-1$ занятых клеток, поэтому для него можно составит систему из m+n-1 независимых уравнений $\alpha_i + \beta_j = c_{ij}$ с m+n неизвестными.

Уравнений на одно меньше, чем неизвестных, поэтому система является неопределенной и одному неизвестному (обычно α_i) присваивают нулевое значение, после этого остальные потенциалы определяются однозначно.

Пусть известен потенциал α_i, тогда $\beta_j = c_{ij} - \alpha_i$, если известен потенциал β_j, то $\alpha_i = c_{ij} - \beta_j$.

Таким образом, для определения неизвестного потенциала от величины c_{ij} надо вычесть известный потенциал.

Так как в нашем примере m = 4, n = 5, то для того, чтобы найденный в таблице 7.7 план был невырожденным, он должен содержать 4 + 5 – 1 = 8 занятых клеток.

В таблице 7.7 всего шесть занятых клеток, следовательно, в две не занятые клетки необходимо поместить нулевые перевозки.

Заполним нулевыми перевозками клетки A_2B_2 и A_2B_4, так как во второй строке среди незанятых клеток они имеют минимальную стоимость перевозок, а именно: 6 и 7. Запишем целевую функцию двойственной задачи, которую необходимо максимизировать.

$$\varphi(\alpha_1,...,\alpha_4,\beta_1,...,\beta_5) = \sum_{i=1}^{4} a_i\alpha_i + \sum_{j=1}^{5} b_j\beta_j = 100\cdot\alpha_1 + 250\cdot\alpha_2 + 200\cdot\alpha_3 + 300\cdot\alpha_4 +$$

$$+ 200\cdot\beta_1 + 200\cdot\beta_2 + 100\cdot\beta_3 + 100\cdot\beta_4 + 250\cdot\beta_5 (\max)$$

при ограничениях:

$$\begin{cases} \alpha_1 + \beta_4 \le 1 \\ \alpha_2 + \beta_1 \le 2 \\ \alpha_2 + \beta_2 \le 7 \\ \alpha_2 + \beta_4 \le 6 \\ \alpha_2 + \beta_5 \le 11 \\ \alpha_3 + \beta_5 \le 2 \\ \alpha_4 + \beta_2 \le 8 \\ \alpha_4 + \beta_3 \le 12. \end{cases}$$

Получили систему из восьми неравенств и девяти неизвестных. Выберем в качестве свободной переменной $\alpha_2 = 0$, тогда потенциалы $\beta_1, \beta_2, \beta_4, \beta_5$ определятся однозначно. Запишем найденные значения потенциалов в таблицу 7.8.

Таблица 7.8 – Построение системы потенциалов

Поставщики, A_i	Потребители, B_j					Запасы
	$B_1,\beta_1=2$	$B_2,\beta_2=7$	$B_3,\beta_3=11$	$B_4,\beta_4=6$	$B_5,\beta_5=11$	a_i

$A_1,\alpha_1=-5$	10	7	4	1 100	4	100
$A_2,\alpha_2=0$	2 200	7 0	10	6 0	11 50	250
$A_3,\alpha_3=-9$	8	5	3	2	2 200	200
$A_4,\alpha_4=1$	11	8 200	12 100	16	13	300
Потребности b_i	200	200	100	100	250	

Проверим выполнение условия $\alpha_i + \beta_j \le c_{ij}$ (4.7) теоремы для незанятых клеток, если условие не выполняется, найдем разность $(\alpha_i + \beta_j) - c_{ij}$ и запишем в соответствующую клетку (таблица7.9).

Таблица 7.9 – Проверка является ли найденный план оптимальным

Поставщики, A_i	Потребители, B_j					Запасы
	$B_1,\beta_1=2$	$B_2,\beta_2=7$	$B_3,\beta_3=11$	$B_4,\beta_4=6$	$B_5,\beta_5=11$	a_i
$A_1,\alpha_1=-5$	10	7	4 2	1 100	4 2	100
$A_2,\alpha_2=0$	2 200	7 0	10 1	6 0	11 50	250
$A_3,\alpha_3=-9$	8	5	3	2	2 200	200
$A_4,\alpha_4=1$	11	8 200	12 100	16	13	300
Потребности b_i	200	200	100	100	250	

Условие оптимальности не выполняется для трех клеток A_1B_3, A_1B_5 и A_2B_3. Разности для этих клеток соответственно равны 2, 2, 1. Выберем клетку, в которую необходимо перераспределить перевозку. Для этого определим $\max[(\alpha_i + \beta_j) - c_{ij}] = \max[2,2,1] = 2$, значит любую из клеток A_1B_3, A_1B_5 можно сделать занятой. Возьмем клетку A_1B_5, так как в этом случае легче построить цикл, отметим ее знаком «+». Клетка A_1B_5 присоединяется к занятым клеткам, которых становиться *m*+*n*. Таким образом, появляется цикл, все вершины которого, за исключением одной, которая находится на клетке, отмеченной знаком «+», лежат на занятых клетках, причем этот цикл единственный.

Находим цикл и, двигаясь от клетки отмеченной знаком «+», поочередно проставляем знаки «–», «+» в клетках с вершинами цикла (таблица 7.10). Находим $\min x_{ij} = \min[100{,}50] = 50$ – минимальное значение среди перевозок, стоящих в вершинах цикла, отмеченных знаком «–».

Таблица 7.10 – Построение цикла

Поставщики, A_i	Потребители, B_j					Запасы
	$B_1,\beta_1=2$	$B_2,\beta_2=7$	$B_3,\beta_3=11$	$B_4,\beta_4=6$	$B_5,\beta_5=11$	a_i
$A_1,\alpha_1=-5$	10	7	4	1 – 100	4 +	100
$A_2,\alpha_2=0$	2	7	10	6 +	11 –	250

	200	0		0	50	
A_3,α_3=-9	8	5	3	2	2 200	200
A_4,α_4=1	11	8 200	12 100	16	13	300
Потребности b_i	200	200	100	100	250	

К перевозкам, стоящим в занятых клетках, прибавляем 50, а из перевозок, стоящих в незанятых клетках, вычитаем 50, получаем новый план (таблица 7.11).

Таблица 7.11 – Новый опорный план

Поставщики, A_i	Потребители, B_j					Запасы a_i
	B_1,β_1=2	B_2,β_2=7	B_3,β_3=11	B_4,β_4=6	B_5,β_5=9	
A_1,α_1=-5	10	7	4 2	1 50	4 50	100
A_2,α_2=0	2 200	7 0	10 1	6 50	11	250
A_3,α_3=-7	8	5	3	2	2 200	200
A_4,α_4=1	11	8 200	12 100	16	13	300
Потребности b_i	200	200	100	100	250	

Пересчитываем потенциалы, так клетка A_2B_5 стала не занятой, а клетка A_1B_5 прибавилась к занятым клеткам. Проверяем является ли найденный план оптимальным. Условие оптимальности не выполняется для двух клеток A_1B_3 и A_2B_3. Разности для этих клеток соответственно равны 2, 1. Выберем клетку, в которую необходимо перераспределить перевозку. Для этого определим $\max[(\alpha_i+\beta_j)-c_{ij}]=\max[2,1]=2$, значит клетку A_1B_3 надо сделать занятой. Отметим ее знаком «+». Клетка A_1B_3 присоединяется к занятым клеткам, которых становиться m+n. Таким образом, появляется цикл все вершины которого, за исключением одной, которая находится на клетке, отмеченной знаком «+», лежат на занятых клетках, причем этот цикл единственный (таблица 7.12). Находим цикл и, двигаясь от клетки отмеченной знаком «+», поочередно проставляем знаки «–», «+» в клетках с вершинами цикла.

Таблица 7.12– Построение цикла

Поставщики, A_i	Потребители, B_j					Запасы a_i
	B_1,β_1=2	B_2,β_2=7	B_3,β_3=11	B_4,β_4=6	B_5,β_5=9	
A_1,α_1=-5	10	7	4 +	1 – 50	4 50	100
A_2,α_2=0	2 200	7– 0	10 1	6 + 50	11	250
A_3,α_3=-7	8	5	3	2	2 200	200
A_4,α_4=1	11	8+	12–	16	13	300

		200	100			
Потребности b_i	200	200	100	100	250	

Определяем $\min x_{ij} = \min[0,100,50] = 0$ – минимальное значение среди перевозок, стоящих в вершинах цикла, отмеченных знаком «–». Значит, нулевую перевозку необходимо переместить в клетку A_1B_3, сделав ее занятой, и опять пересчитать потенциалы (таблица 7.11).

Проверяем найденный план на оптимальность. Для всех незанятых клеток условие оптимальности выполняется. Значит, найденный план является оптимальным.

Находим значение целевой функции.

Таблица 7.13 – Построение новой системы потенциалов

Поставщики, A_i	Потребители, B_j					Запасы a_i
	$B_1,\beta_1=2$	$B_2,\beta_2=5$	$B_3,\beta_3=9$	$B_4,\beta_4=6$	$B_5,\beta_5=9$	
$A_1,\alpha_1=-5$	10	7	4 0	1 50	4 50	100
$A_2,\alpha_2=0$	2 200	7	10	6 50	11	250
$A_3,\alpha_3=-7$	8	5	3	2	2 200	200
$A_4,\alpha_4=3$	11	8 200	12 100	16	13	300
Потребности b_i	200	200	100	100	250	

$F_{\min} = 1\cdot50+4\cdot50+6\cdot50+2\cdot200+2\cdot200+8\cdot200+12\cdot100=4150.$

$\Psi_{\max}=-5\cdot100+0\cdot250-7\cdot200+3\cdot300+2\cdot200+5\cdot200+9\cdot100+6\cdot100+9\cdot250 =4150.$

Так как $F_{\min}=\Psi_{\max} = 4150$, т.е. целевая функция и прямой, и двойственной задачи имеет одно и тоже значение и выполняется условие оптимальности для незанятых клеток таблицы 4.13, значит, найденный план является оптимальным.

Ответ: $F_{\min}= 4150$ при: $x_{14} = 50$, $x_{15} =50$, $x_{12} =200$, $x_{14} =50$, $x_{35} =200$, $x_{42} =200$, $x_{43} =100$.

Проверим полученный результат, решив задачу с помощью инструмента «Поиск решения» табличного процессора Excel.

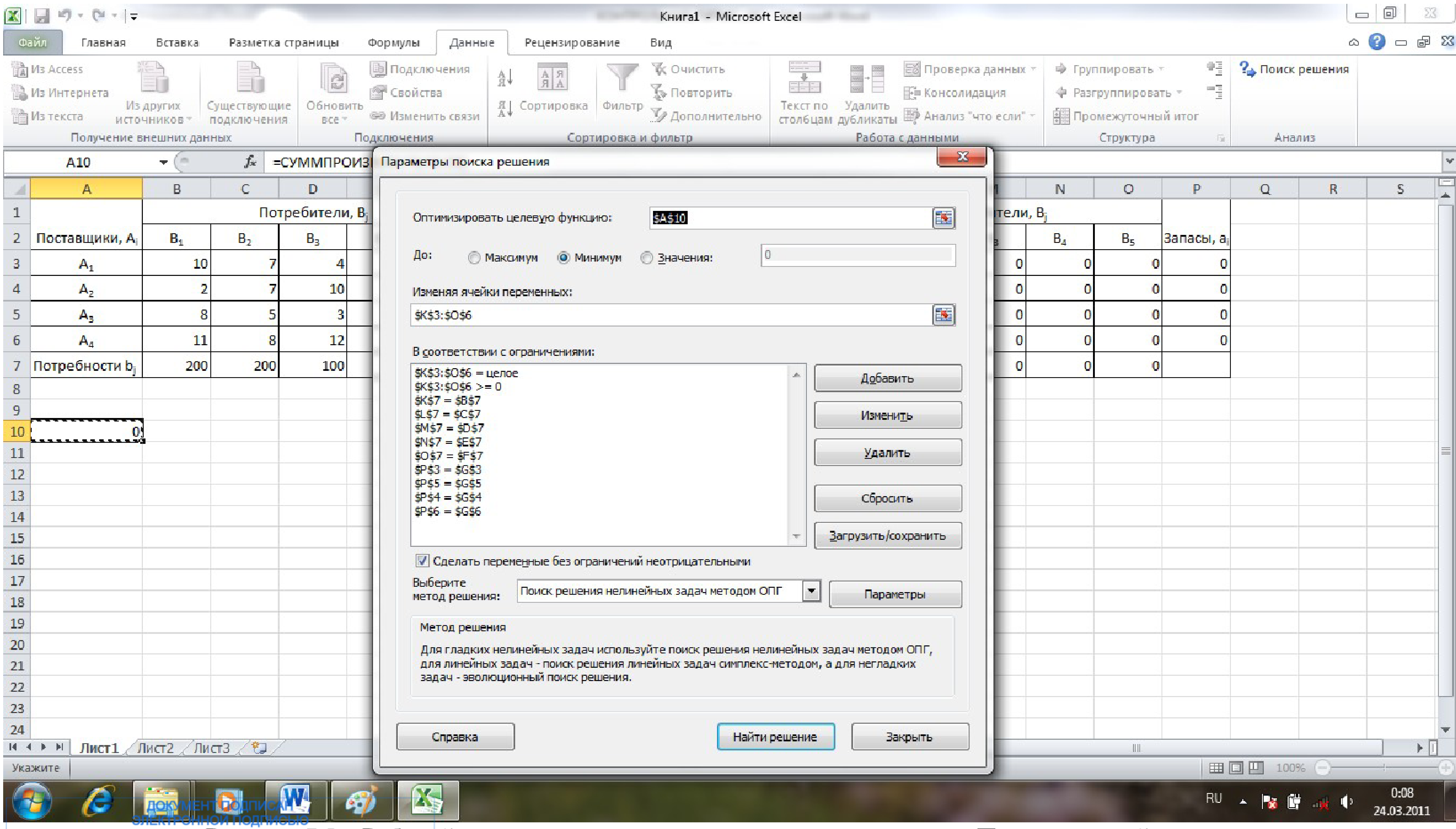


Рисунок 7.5 – Рабочий лист с введенными данными и окно диалога «Поиск решений»

Рабочий лист с решением задачи, представленной матрицей планирования (таблица 7.2), в системе компьютерной математики Mathcad представлен на рисунке 7.6.

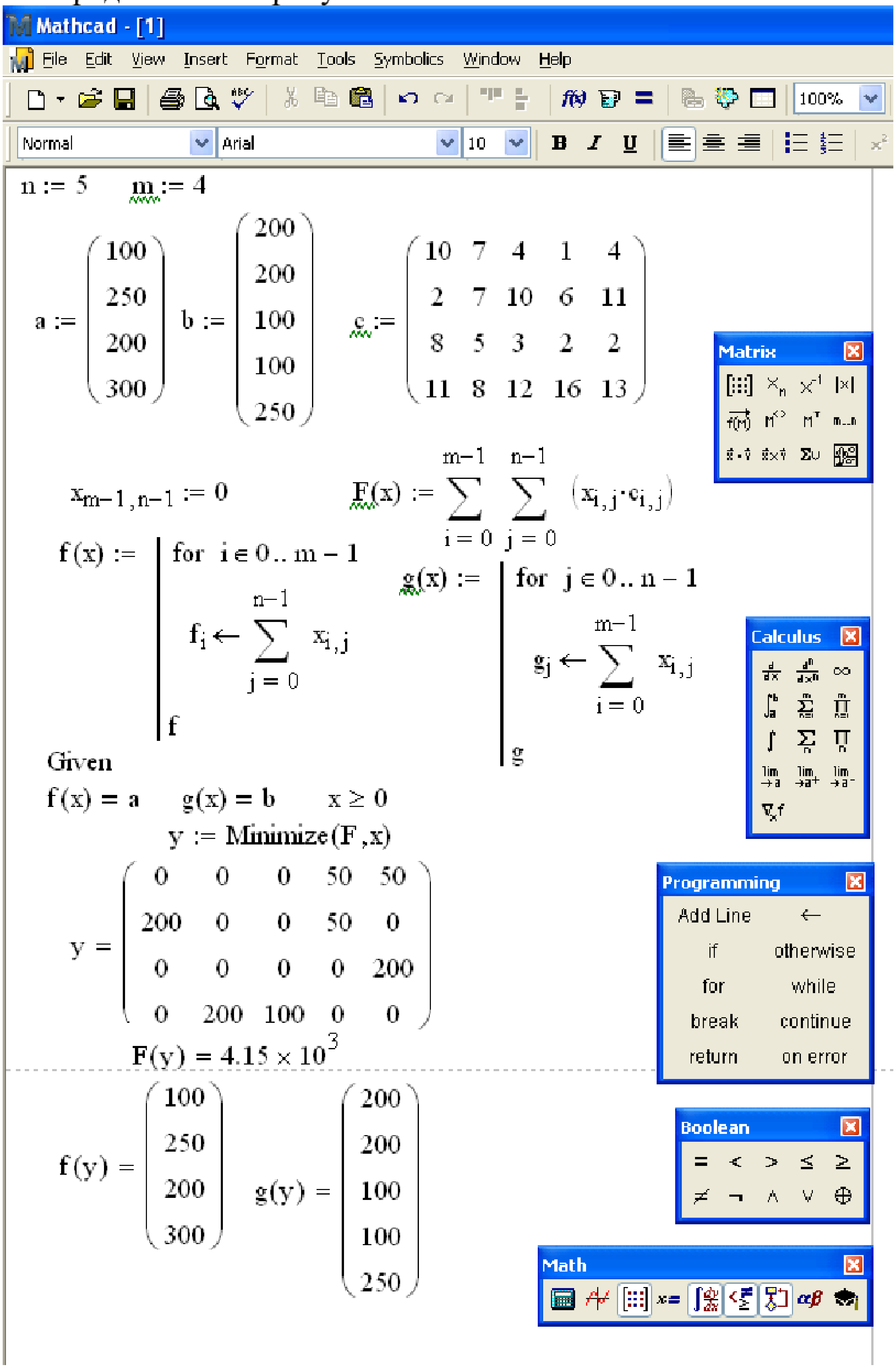


Рисунок 7.6 – Алгоритм решения задачи в Mathcad

Аппаратура и материалы. Для выполнения лабораторной работы необходимо использовать следующее: аппаратное обеспечение: персональный компьютер и программное обеспечение: операционную систему Windows 10 и выше; Microsoft Office13 и выше, системы компьютерной математики Machcad и MATLAB R2017a и выше.

Указания по технике безопасности. Самостоятельно не производить установку и удаление программного обеспечения; ремонт персонального компьютера. Соблюдать правила технической эксплуатации и техники безопасности при работе с электрооборудованием.

Методика и порядок выполнения работы

Выполните предложенные задания.

Задание 7.1. Информационные модели принятия решений в условиях определенности.

Найти первоначальный опорный план (методами: северо-западного угла; минимальной стоимости; двойного предпочтения) следующей задачи**:** Пусть имеется три пункта отправления (поставщика) A1, A2, A3 и четыре пункта назначения (потребителя) B1, B2, B3, B4. Известны расстояния (в км) от каждого поставщика до каждого потребителя. Эти расстояния приведены ниже в таблице 7.14 в левом верхнем углу клеток. Требуется составить такой план перевозок груза, при котором общий грузооборот (в т.км) был бы минимальным.

Таблица 7.14 – Матрица стоимостей

Поставщики	Потребители				Запасы
	B_1	B_2	B_3	B_4	a_i
A_1	6	12	20	22	1500
A_2	10	18	18	17	2800
A_3	1	4	6	11	1700
Потребности *bj*	1830	1160	1610	1400	

1. Постройте математическую модель транспортной задачи.
2. Найдите опорный план методами: северо-западного угла; минимальной стоимости; двойного предпочтения.
3. Постройте систему потенциалов и проверьте найденные планы на оптимальность.
4. Найдите оптимальный план:
   - с помощью инструмента «Поиск решения» табличного процессора Excel;
   - с помощью системы компьютерной математики Mathcad;
   - методом потенциалов, взяв за первоначальный любой из опорных планов, не являющихся оптимальным.
5. Сделайте вывод.

Задание 6. 2. Индивидуальное задание: Используя таблицу из своего варианта (таблица 7.15), постройте математическую модель заданной экономической системы. Постройте опорный план методами: северо-западного угла; минимальной стоимости; двойного предпочтения, найдите оптимальный план с помощью инструмента «Поиск инструмента».

Таблица 7.15 – Варианты первого задания

Вариант №	Условие задачи					
1.	Поставщики	Потребители				Запасы
		B_1	B_2	B_3	B_4	a_i
	A_1	5	2	8	3	120
	A_2	7	3	9	4	180
	A_3	3	2	1	2	160
	Потребности b_j	40	160	140	120	
2.	Поставщики	Потребители				Запасы
		B_1	B_2	B_3	B_4	a_i
	A_1	3	2	1	3	210
	A_2	8	7	9	10	190
	A_3	4	2	10	12	100
	Потребности b_j	130	100	150	120	
3.	Поставщики	Потребители				Запасы
		B_1	B_2	B_3	B_4	a_i
	A_1	5	2	8	3	120
	A_2	7	3	9	4	180
	A_3	3	2	1	2	150
	Потребности b_j	130	100	170	50	
4.	Поставщики	Потребители				Запасы
		B_1	B_2	B_3	B_4	a_i
	A_1	2	1	4	3	120
	A_2	6	4	3	7	170
	A_3	4	2	5	9	110
	Потребности b_j	50	00	150	110	
5.	Поставщики	Потребители				Запасы
		B_1	B_2	B_3	B_4	a_i
	A_1	5	2	8	3	110
	A_2	7	3	9	4	180
	A_3	3	2	1	2	210
	Потребности b_j	140	100	140	120	
6.	Поставщики	Потребители				Запасы
		B_1	B_2	B_3	B_4	a_i
	A_1	5	2	8	3	100
	A_2	7	3	9	4	160
	A_3	3	2	1	2	140
	Потребности b_j	100	200	50	50	
7.	Поставщики	Потребители				Запасы
		B_1	B_2	B_3	B_4	a_i
	A_1	3	2	1	4	110
	A_2	9	6	4	5	150
	A_3	6	2	2	7	140

	Потребности b_j	20	140	130	110	
8.	Поставщики	Потребители				Запасы
		B_1	B_2	B_3	B_4	a_i
	A_1	4	2	4	3	130
	A_2	7	4	9	4	170
	A_3	11	12	3	10	120
	Потребности b_j	70	140	110	100	
9.	Поставщики	Потребители				Запасы
		B_1	B_2	B_3	B_4	a_i
	A_1	1	5	6	3	150
	A_2	7	4	3	8	100
	A_3	8	2	3	1	150
	Потребности b_j	180	60	40	120	
10.	Поставщики	Потребители				Запасы
		B	B_2	B_3	B_4	a_i
	A_1	1	5	6	3	120
	A_2	5	9	3	4	180
	A_3	3	2	7	2	100
	Потребности b_j	80	60	140	120	
11.	Поставщики	Потребители				Запасы
		B_1	B_2	B_3	B_4	a_i
	A_1	2	3	2	7	100
	A_2	1	3	9	4	250
	A_3	2	6	3	1	100
	Потребности b_j	80	160	140	120	
12.	Поставщики	Потребители				Запасы
		B_1	B_2	B_3	B_4	a_i
	A_1	2	7	5	1	100
	A_2	4	2	3		190
	A_3	9	6	2	8	150
	Потребности b_j	80	120	100	140	
13.	Поставщики	Потребители				Запасы
		B_1	B_2	B_3	B_4	a_i
	A_1	1	4	8	6	140
	A_2	8	2	4	5	150
	A_3	5	1	3	7	150
	Потребности b_j	80	120	140	100	
14.	Поставщики	Потребители				Запасы
		B_1	B_2	B_3	B_4	a_i
	A_1	2	7		7	140
	A_2	7	1	8	3	170
	A_3	5	2	5	1	150
	Потребности b_j	80	100	140	140	
15.	Поставщики	Потребители				Запасы
		B_1	B_2	B_3	B_4	a_i
	A_1	3	8	2	7	140
	A_2	7	2	7	1	190
	A_3	4	3	5	4	110
	Потребности b_j	40	120		140	

№	Поставщики	Потребители				Запасы
				40		
16.	Поставщики	Потребители				Запасы
		B_1	B_2	B_3	B_4	a_i
	A_1	3	7	2	7	140
	A_2	7	3	1	3	190
	A_3	5	2	5	4	150
	Потребности b_j	80	120	140	140	
17.	Поставщики	Потребители				Запасы
		B_1	B_2	B_3	B_4	a_i
	A_1	3	7	2	7	140
	A_2	1	8	3	9	150
	A_3	5	1	2	1	150
	Потребности b_j	80	100	120	140	
18.	Поставщики	Потребители				Запасы
		B_1	B_2	B_3	B_4	a_i
	A_1	1	6	3	3	140
	A_2	5	2	4	1	240
	A_3	4	8	6	5	150
	Потребности b_j	130	120	140	140	
19.	Поставщики	Потребители				Запасы
		B_1	B_2	B_3	B_4	a_i
	A_1	2	1	3	5	100
	A_2	7	3	1	3	190
	A_3	5	7	6	4	190
	Потребности b_j	100	120	120	140	
20.	Поставщики	Потребители				Запасы
		B_1	B_2	B_3	B_4	a_i
	A_1	3	8	2	7	100
	A_2	1	10	3	9	170
	A_3	5	11	2	1	170
	Потребности b_j	120	100	120	100	

Содержание отчета и его форма

Подготовьте отчет, в котором опишите технологию решения задач, используя задания своего варианта. Отчет по лабораторной работе должен содержать:

1) название работы;
2) цель лабораторной работы;
3) условия выполненных заданий и их решение;
4) выводы;
5) ответы на контрольные вопросы.

Вопросы для защиты работы:

1. Сформулировать транспортную задачу в общем виде.
2. Сформулировать алгоритм нахождения первоначального опорного плана транспортной задачи методом северо-западного угла.
3. Сформулировать алгоритм нахождения первоначального опорного плана транспортной задачи методом минимальной стоимости.

4. Сформулировать алгоритм нахождения первоначального опорного плана транспортной задачи методом двойного предпочтения.
5. Как проверить найденный план на оптимальность?

ЛАБОРАТОРНАЯ РАБОТА 8
Элементы теории игр в задачах моделирования, поиск оптимальной смешанной стратегии

Цель и содержание: Научиться строить и анализировать модели задач принятия решения в условиях неопределенности.

Организационная форма занятий: решение проблемных задач, разбор конкретных ситуаций

Вопросы для обсуждения на лабораторном занятии: построение, решение и анализ моделей задач теории игр.

Формируемые компетенции: ОПК-7.1 ОПК-7.2 ОПК-8.1 ОПК-8.2

Теоретическое обоснование

Изучите теоретический материал по данной теме, используя материал, изложенный ниже.

Теория игр может применяться в различных сферах искусственного интеллекта:

- Мультиагентные системы искусственного интеллекта.
- Имитация и Усиление обучения.
- Обучение противника в Генеративных Состязательных Сетях (GAN).

Теория игр может также использоваться для описания многих ситуаций в нашей повседневной жизни и моделях машинного обучения

Игры занимают ключевое место в эволюции искусственного интеллекта (ИИ), так как в настоящее время игровые среды становятся популярным тренировочным механизмом в таких областях, как обучение с подкреплением (reinforcement learning) и имитационное обучение (imitation learning). В контексте искусственного интеллекта и систем глубокого обучения теория игр обеспечивает некоторые ключевые возможности, необходимые в многоагентных средах, в которых различные программы ИИ должны взаимодействовать или соревноваться друг с другом для достижения какой-либо цели.

В теории, любая многоагентная система ИИ может работать с геймифицированными взаимодействиями участников. Раздел математики, который формулирует игровые принципы, называется теорией игр. [https://medium.datadriveninvestor.com/game-theory-for-data-scientists-e44048a7e935]

Теорией игр называют математическую дисциплину, занимающуюся обоснованием оптимальных решений, принимаемых в тех или иных конфликтных ситуациях. Под ***конфликтной ситуацией*** понимается такая ситуация, в которой необходимо принимать решения в условиях, когда целям одной стороны противостоят противоположные цели другой стороны. Подобные ситуации характерны для военных действий, игровых видов спорта, практической деятельности. Во всех подобных случаях происходит столкновение противоположных интересов, принятие решения каждой из сторон связано с преодолением конфликта. Принятие решения в конфликтной ситуации затрудняется из-за неопределенности поведения противника. Теория игр позволяет точным математическим языком описать правила игры, дать формулы, оценивающие ее ситуации и указывающие правильный путь к победе. Эта теория позволяет с некоторой долей точности учесть многие обстоятельства, влияющие на борьбу и предсказать ее исход с определенной степенью достоверности.

Основные понятия теории игр. Классификация и описание игр

В основе теории игр лежит ***конфликт***, который выступает как нечто формализуемое, объективно анализируемое и просчитываемое при необходимости достижения цели. Для математического анализа конфликта строится упрощенная модель конфликтной ситуации, которую называют ***игрой***. Ее участников по традиции называют ***игроками.*** Игру ведут по определенным правилам. Результат игры называют ***выигрышем*** или ***платежом***. Одну реализацию игры – ***партией.*** Выбор игроком того или иного действия – ***ходом.*** Различают два вида ходов – личные и случайные. ***Личный ход*** предполагает сознательный выбор игроком того или иного действия, разрешенного правилами игры. ***Случайный ход*** не зависит от воли игрока и является результатом того или иного случайного процесса (например, сдача карт). Игры, состоящие только из случайных ходов, называются ***азартными***. Игры, в которых имеются личные ходы, называются ***стратегическими***. Существуют также стратегические игры, состоящие, как из личных, так и случайных ходов. Теория игр занимается стратегическими играми. Каждый игрок располагает конечным или бесконечным набором допустимых решений, называемых ***стратегиями***. Выигрыш каждого игрока определяется его платежной функцией, значения которой зависят от стратегий всех участников игры.

Фактически игра представляет собой совокупность правил, известных всем игрокам. Эти правила, с одной стороны, определяют множества стратегий игроков, а с другой – последствия и выигрыши в результате выбора каждой из стратегий. Задача теории игр определить такую стратегию игрока, при которой его шансы на выигрыш оказались бы наибольшими.

В теории игр рассматриваются игры двух и игры многих лиц, игры, в которых каждый игрок может делать лишь один ход, и игры с множеством ходов. Стратегии игроков могут использовать или не использовать случайный механизм. В конфликтных ситуациях, в которых допустимо применение случайного механизма, последствия игры определяются средним выигрышем при многократном повторении игры. В таких играх выбор стратегии предусматривает также выбор функции распределения используемого случайного механизма.

Классификацию игр проводят по различным признакам:

- по числу игроков;
- по числу стратегий;
- по свойствам платежной функции;
- по характеру предварительной договоренности между игроками.

Игру, в которой участвует *n* игроков, называют ***игрой с n участниками.*** В игре могут возникать как конфликтные ситуации, так и необходимость в координированных действиях ***(кооперация).*** Если в игре участвует не менее трех игроков, то могут создаваться ***коалиции,*** т.е. группы из двух или более игроков, имеющих общую цель и координирующих свои стратегии.

По количеству стратегий различают ***игры конечные* и *бесконечные.*** Если хотя бы один из игроков располагает бесконечным множеством стратегий, то игру называют бесконечной. Если же каждый из игроков располагает конечным множеством стратегий, то игру называют конечной.

Еще один способ классификации игр – по свойствам платежной функции. В ***игре с нулевой суммой*** общая сумма выигрышей всех игроков равна нулю. В игре с нулевой суммой и двумя участниками выигрыш одного из них равен проигрышу другого. Таким образом, в играх с нулевой суммой существует конфликт между игроками и поэтому их называют также ***антагонистическими играми.*** В общем случае в игре с нулевой суммой, как правило, имеют место и конфликты, и согласованные действия игроков. Прямой противоположностью играм с нулевой суммой являются ***игры*** двух игроков ***с постоянной разностью,*** в которых оба игрока выигрывают или проигрывают одновременно. Поэтому игрокам выгодно действовать согласованно.

В зависимости от характера предварительной договоренности между игроками различают ***кооперативные* и *некооперативные игры.*** Игра кооперативная, если до ее начала игроки образуют коалиции и принимают взаимообязывающие соглашения о координации своих стратегий. В противном случае игра будет некооперативной.

Существуют два основных способа описания и анализа любой конкретной игры.

Первый способ предполагает:

- перечисление ходов, которые могут делать игроки;
- определение информации, которой располагают игроки в процессе игры;
- определение возможных вариантов действий игроков;
- указание предельных размеров платежей в конце игры.

Игру, описанную подобным образом, называют ***игрой в развернутой*** или ***экстенсивной форме,*** а само описание, как правило, составляют в виде ***дерева игры***. Игры в развернутой форме называют также ***позиционными играми.***

Игру в развернутой форме называют ***игрой с полной информацией***, если в ней нельзя делать одновременно несколько ходов и если участникам известны выборы, сделанные при предшествующих ходах, включая и случайные ходы. Примером игры с полной информацией являются шахматы. Покер представляет собой ***игру с неполной информацией,*** так как игрокам неизвестно, какие карты находятся на руках у противника.

Второй способ описания игры предполагает рассмотрение всех возможных стратегий каждого игрока и определение платежей, соответствующих любым возможным комбинациям стратегий всех игроков. Игру, описанную вторым способом, обычно называют игрой ***в нормальной форме.*** Естественно, зная развернутую форму игры, всегда можно представить ее в нормальной форме.

Нормальная форма игры двух участников состоит из двух матриц**,** содержащих суммы выигрышей и проигрышей каждого из игроков для любой из возможных пар стратегий. Обычно эти две матрицы объединяются в одну, называемую ***платежной матрицей игры.***

Платежная матрица игры. Нижняя и верхняя цена игры. Принцип минимакса (максимина)

Рассмотрим платежную матрицу игры на примере простейшей игры. Простейшие игры – это игры двух лиц с нулевой суммой, т.е. ***парные игры***, в которых выигрыш одного игрока совпадает с проигрышем другого.

Результаты конечной игры (парной) с нулевой суммой можно задавать матрицей, строки и столбцы которой соответствуют различным отражениям, а ее элементы – соответствующие выигрыши одной стороны (равные проигрышам другой стороны). Эта матрица называется платежной матрицей или матрицей игры. При этом удобно проигрыш первой стороны рассматривать как ее отрицательный выигрыш, а выигрыш второй – как ее отрицательный проигрыш. Если первая сторона имеет m стратегий, а вторая n, то говорят, что мы имеем дело с игрой m x n. Итак, пусть первый игрок имеет m стратегий $i = 1,2,..., m$, второй имеет n стратегий $j = 1,2,..., n$. Каждой паре стратегий (i,j) поставлено в соответствие число a_{ij}, выражающее выигрыш первого игрока за счёт второго игрока, если первый игрок примет свою i-ю стратегию, а второй – свою j-ю стратегию. В случае выигрыша первого игрока – $a_{ij} > 0$, а в случае его проигрыша – $a_{ij} < 0$.

Каждая стратегия игрока $i=\overline{1,m}$; $j = \overline{1,n}$ часто называется чистой стратегией.

Если рассмотреть матрицу:

$$A = \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1j} & \cdots & a_{1n} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ a_{i1} & a_{i2} & \cdots & a_{ij} & \cdots & a_{in} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ a_{m1} & a_{m2} & \cdots & a_{mj} & \cdots & a_{mn} \end{pmatrix},$$

то проведение каждой партии матричной игры с матрицей A сводится к выбору первым игроком i-й строки, а вторым игроком j-го столбца и получения первым игроком (за счёт второго игрока) выигрыша a_{ij}.

Главным в исследовании игр является понятие оптимальных стратегий игроков. В это понятие интуитивно вкладывается такой смысл: стратегия игрока является оптимальной, если применение этой стратегии обеспечивает ему наибольший гарантированный выигрыш при всевозможных стратегиях другого игрока. Исходя из этих позиций, первый игрок исследует матрицу выигрышей A следующим образом: для каждого значения i ($i = \overline{1,m}$) определяется минимальное значение выигрыша в зависимости от применяемых стратегий второго игрока $\min\limits_{j} a_{ij}$, ($i = \overline{1,m}$), т.е. определяется минимальный выигрыш для первого игрока при условии, что он примет свою i-ю чистую стратегию, затем из этих минимальных выигрышей отыскивается такая стратегия, при которой этот минимальный выигрыш будет максимальным, т.е. находится максимин: $\max\limits_{i} \min\limits_{j} a_{ij} = \underline{\alpha}$ – нижняя цена игры.

Число $\underline{\alpha}$, называется ***нижней ценой игры*** и показывает, какой минимальный выигрыш может гарантировать себе первый игрок, применяя свои чистые стратегии при всевозможных действиях второго игрока.

Второй игрок при оптимальном своём поведении должен стремиться по возможности за счёт своих стратегий максимально уменьшить выигрыш первого игрока. Поэтому для второго игрока отыскивается $\max\limits_{i} a_{ij}$, т.е. определяется max выигрыш первого игрока при условии, что второй игрок применит свою j-ю чистую стратегию, затем второй игрок отыскивает такую свою стратегию, при которой первый игрок получит min выигрыш, т.е. находит минимакс: $\min\limits_{j} \max\limits_{i} a_{ij} = \overline{\alpha}$.

Число $\overline{\alpha}$ называется ***верхней ценой игры*** и показывает, какой максимальный выигрыш за счёт своих стратегий может себе гарантировать первый игрок.

Другими словами, применяя свои чистые стратегии, первый игрок может обеспечить себе выигрыш не меньше $\underline{\alpha}$, а второй игрок за счёт применения своих чистых стратегий может не допустить выигрыш первого игрока больше, чем $\overline{\alpha}$.

Стратегия, соответствующая минимаксу, называется минимаксной стратегией.

Принцип, диктующий игрокам выбор наиболее «осторожных» минимаксной и максиминной стратегий, называется принципом минимакса.

Если в игре с матрицей А $\underline{\alpha} = \overline{\alpha}$, то говорят, что эта игра имеет седловую точку**,** т.е**.** решение в чистых стратегиях и цену игры $\upsilon = \underline{\alpha} = \overline{\alpha}$.

Седловая точка – это пара чистых стратегий соответственно первого и второго игроков, при которых достигается равенство $\underline{\alpha} = \overline{\alpha}$. В это понятие вложен следующий смысл: если один из игроков придерживается стратегии, соответствующей седловой точке, то другой игрок не сможет поступить лучше, чем придерживаться стратегии, соответствующей седловой точке.

Таким образом, седловой элемент является минимальным в i-й строке и максимальным в j-м столбце в матрице А. Отыскание седловой точки матрицы А происходит следующим образом: в матрице А последовательно в каждой строке находят минимальный элемент и проверяют, является ли этот элемент максимальным в своём столбце. Если да, то он и есть седловой элемент, а пара стратегий, ему соответствующая, образует седловую точку. Пара чистых стратегий первого и второго игроков, образующая седловую точку и седловой элемент a_{ij} , называется ***решением игры***.

Пример 1

$$
\begin{array}{ll}
 & \min\limits_j a_{ij} \\
 & \quad \| \\
A = \begin{pmatrix} 5 & 1 & 2 \\ 4 & 9 & 8 \\ 6 & 9 & 6 \end{pmatrix} & \left.\begin{matrix} 1 \\ 4 \\ 6 \end{matrix}\right\} \max\limits_i \min\limits_j a_{ij} = 6 \\
\max\limits_i a_{ij} = \underbrace{6 \quad 9 \quad 8}_{\min\limits_j \max\limits_i a_{ij} = 6.} &
\end{array}
$$

Седловой точкой является пара (i = 3; j = 1), при которой $\upsilon = \underline{\alpha} = \overline{\alpha} = 6$.

Заметим, что, хотя выигрыш в ситуации (3;3) также равен 6, однако она не является седловой точкой, т.к. этот выигрыш не является максимальным среди выигрышей третьего столбца.

Пример 2

$$
H=\begin{array}{c} \\ \\ \max\limits_i a_{ij} \end{array}\begin{pmatrix} 1 & 3 \\ 4 & 2 \end{pmatrix}\begin{array}{c} \min\limits_j a_{ij} \\ \to \;\; 1 \\ \to \;\; 2 \end{array}\Bigg\}\max_i \min_j a_{ij}=2
$$

$$
\underbrace{4 \quad 3}
$$

$$
\min_j \max_i a_{ij}=3.
$$

Из анализа матрицы выигрышей видно, что $\underline{\alpha}<\overline{\alpha}$, т.е. данная матрица не имеет седловой точки.

Итак, исследование в матричных играх начинается с нахождения её седловой точки в чистых стратегиях. Если матричная игра имеет седловую точку в чистых стратегиях, то нахождением этой седловой точки заканчивается исследование игры. Если же в игре нет седловой точки в чистых стратегиях, то можно найти нижнюю и верхнюю чистые цены этой игры, которые указывают, что первый игрок не должен надеяться на выигрыш больший, чем верхняя цена игры, и может быть уверен в получении выигрыша не меньше нижней цены игры. Улучшение решений матричных игр следует искать в использовании секретности применения чистых стратегий и возможности многократного повторения игр в виде партии. Этот результат достигается путём применения чистых стратегий случайно, с определённой вероятностью.

В игре двух лиц с нулевой суммой и конечным числом стратегий у каждого игрока под оптимальными стратегиями естественно понимать стратегии, при которых достигается максимально возможный гарантированный выигрыш в наименее благоприятных условиях. Этот принцип называется принципом максимина. В игре двух лиц с нулевой суммой существование оптимальной (максимальной) стратегии в каждой отдельной партии гарантируются только в играх с полной информацией. В игре с неполной информацией максимальная стратегия обеспечивается лишь при применении случайного механизма для выбора очередного хода и многократном повторении игры. В этом случае говорят об использовании смешанных стратегий.

В более сложных играх поиск оптимальных стратегий поведения связан с понятием рационального, справедливого решения игры (так называемый принцип в оптимальной игре), определяемого в зависимости от конкретных содержательных особенностей конфликта.

Принципы оптимальности в теории игр вводятся как некоторые правдоподобные описания представления о фактическом поведении игроков, принимающих решения в условиях конфликта или неопределенности. Все предложенные до сих пор принципы оптимальности прямо или косвенно отражают идею устойчивости ситуации, удовлетворяющей этим принципам. Однако в различных принципах оптимальности интуитивно оправданная идея устойчивости реализуется по-разному.

Многие принципы оптимального поведения игроков, представляющихся на первый взгляд вполне разумными, в действительности могут оказаться нереализуемыми и поэтому неприменимыми. В связи с этим изучение условий, при которых те или иные интуитивно оправданные принципы оптимальности оказываются реализуемыми, и доказательства соответствующих теорем существования решения игры являются центральной проблемой теории игр, тесно связанной с проблемой выбора из множества допустимых альтернатив и проблемой группового выбора.

Поиск оптимальной смешанной стратегии. Связь теории с линейным программированием

Так как каждый раз применение игроком одной чистой стратегии исключает применение другой, то чистые стратегии являются несовместными событиями. Кроме того, они являются единственными возможными событиями.

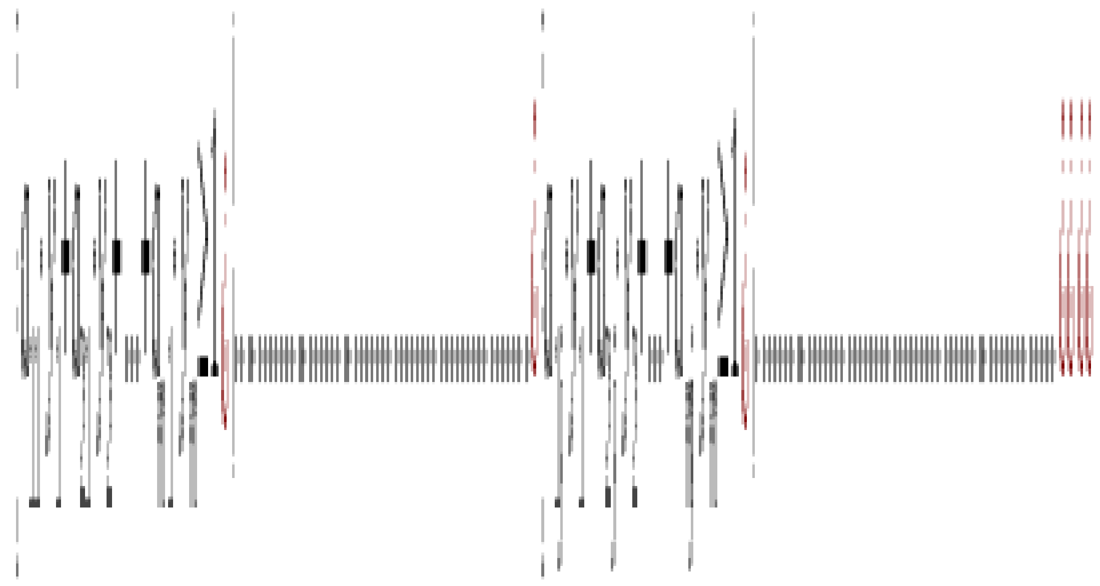
Смешанной стратегией игрока называется полный набор вероятностей применения его чистых стратегий.

Рассмотрим поиск оптимальной смешанной стратегии на примере матрицы, приведенной выше. Пусть первый игрок имеет m чистых стратегий. Для получения наибольшего эффекта он должен использовать все или некоторые из этих стратегий случайным образом, но не с одинаковыми, а с разными (специально вычисленными) вероятностями. Пусть первая стратегия используется с вероятностью p_1, вторая с вероятностью p_2 и т.д. Говорят, что первый игрок применил свою смешанную стратегию $S_1(p_1, p_2, \ldots, p_m)$. Аналогично, смешанная стратегия второго игрока, который имеет n чистых стратегий – $S_2(q_1, q_2, \ldots, q_n)$.

Для определения оптимальной смешанной стратегии первого игрока необходимо найти вероятности: $p_1, p_2, \ldots, p_m$ применения соответствующих чистых стратегий и цену игры ν, которая больше нижней цены игры, но меньше верхней цены игры. При этом $\sum_{i=1}^{m} p_i = 1$.

Чистая стратегия есть частный случай смешанной стратегии. Действительно, если в смешанной стратегии какая-либо i-я чистая стратегия применяется с вероятностью 1, то все остальные чистые стратегии не применяются. И эта i-я чистая стратегия является частным случаем смешанной стратегии. Для соблюдения секретности каждый игрок применяет свои стратегии независимо от выбора другого игрока.

Допустим, второй игрок выбирает свою первую чистую стратегию, тогда средний выигрыш первого игрока будет равен: $a_{11} \cdot p_1 + a_{21} \cdot p_2 + \ldots + a_{i1} \cdot p_i + \ldots + a_{m1} \cdot p_m \geq v$, используем первый столбец платежной матрицы. Если второй игрок выбирает вторую чистую стратегию, то средний выигрыш первого игрока составит: $a_{12} \cdot p_1 + a_{22} \cdot p_2 + \ldots + a_{i2} \cdot p_i + \ldots + a_{m2} \cdot p_m \geq v$, используем второй столбец платежной матрицы и т.д., всего получится система, состоящая из n неравенств. Разделим каждое неравенство системы и уравнение $\sum_{i=¿1}^{m} p_i ¿$ = 1 на v. Введем обозначения: $y_1 = \frac{p_1}{v}, y_2 = \frac{p_2}{v}, \ldots, y_m = \frac{p_m}{v}$, в результате получим:



и $y_1 + y_2 + \ldots + y_m = \frac{1}{v}$. Для того, чтобы цена игры v была как можно больше необходимо, чтобы 1/v была как можно меньше.

Поскольку первый игрок стремится найти такие значения y_i и, следовательно, p_i , чтобы цена игры υ была максимальной, то решение первой задачи сводится к нахождению таких неотрицательных значений y_i $\left(i=\overline{1,m}\right)$, при которых $\sum_{i=1}^{m} y_i \rightarrow \min$, $\sum_{i=1}^{m} a_{ij} y_i \geq 1$.

Поскольку второй игрок стремится найти такие значения x_j и, следовательно, q_j, чтобы цена игры υ была наименьшей, то решение второй задачи сводится к нахождению таких неотрицательных значений x_j , $\left(j=\overline{1,n}\right)$, при которых $\sum_{j=¿1}^{n} x_j \rightarrow \max ¿$, $\sum_{j=¿1}^{n} a_{ij} x_j \leq 1 ¿$.

Таким образом, поиск оптимальной смешанной стратегии первого игрока свелся к решению следующей задачи линейного программирования:

$$\varphi = \sum_{i=¿1}^{m} y_i (\min) ¿$$

при ограничениях:

$$\sum_{i=1}^{m} a_{ij} \cdot y_i \geq 1 \left(j=\overline{1,n}\right)$$

$$y_i \geq 0 \left(i=\overline{1,m}\right).$$

Аналогично для второго игрока:

$$F = \sum_{j=¿1}^{n} x_i (\max) ¿$$

при ограничениях:

$$\sum_{j=1}^{n} a_{ij} \cdot x_j \le 1 \left(i = \overline{1,m} \right)$$

$$x_j \ge 0 \left(j = \overline{1,n} \right).$$

Найдя решения полученных задач линейного программирования, вычисляют цену игры и соответствующие вероятности по формулам:

$$\upsilon = \frac{1}{\sum_{i=1}^{m} y_i} \quad \text{или} \quad \upsilon = \frac{1}{\sum_{j=1}^{n} x_j}; \quad \frac{y_i}{\upsilon} = p_i \quad \left(i = \overline{1,m} \right), \quad \frac{x_j}{\upsilon} = q_j \quad \left(j = \overline{1,n} \right).$$

При решении произвольной конечной игры размера *m* x *n* рекомендуется придерживаться следующей схемы:

1. Исключить из платежной матрицы заведомо невыгодные стратегии по сравнению с другими стратегиями. Такими стратегиями для первого игрока (второго игрока) являются те, которым соответствуют строки (столбцы) с элементами, заведомо меньшими (большими) по сравнению с элементами других строк (столбцов).

2. Определить верхнюю и нижнюю цены игры и проверить, имеет ли игра седловую точку. Если седловая точка есть, то соответствующие ей стратегии игроков будут оптимальными, а цена совпадает с верхней (нижней) ценой.

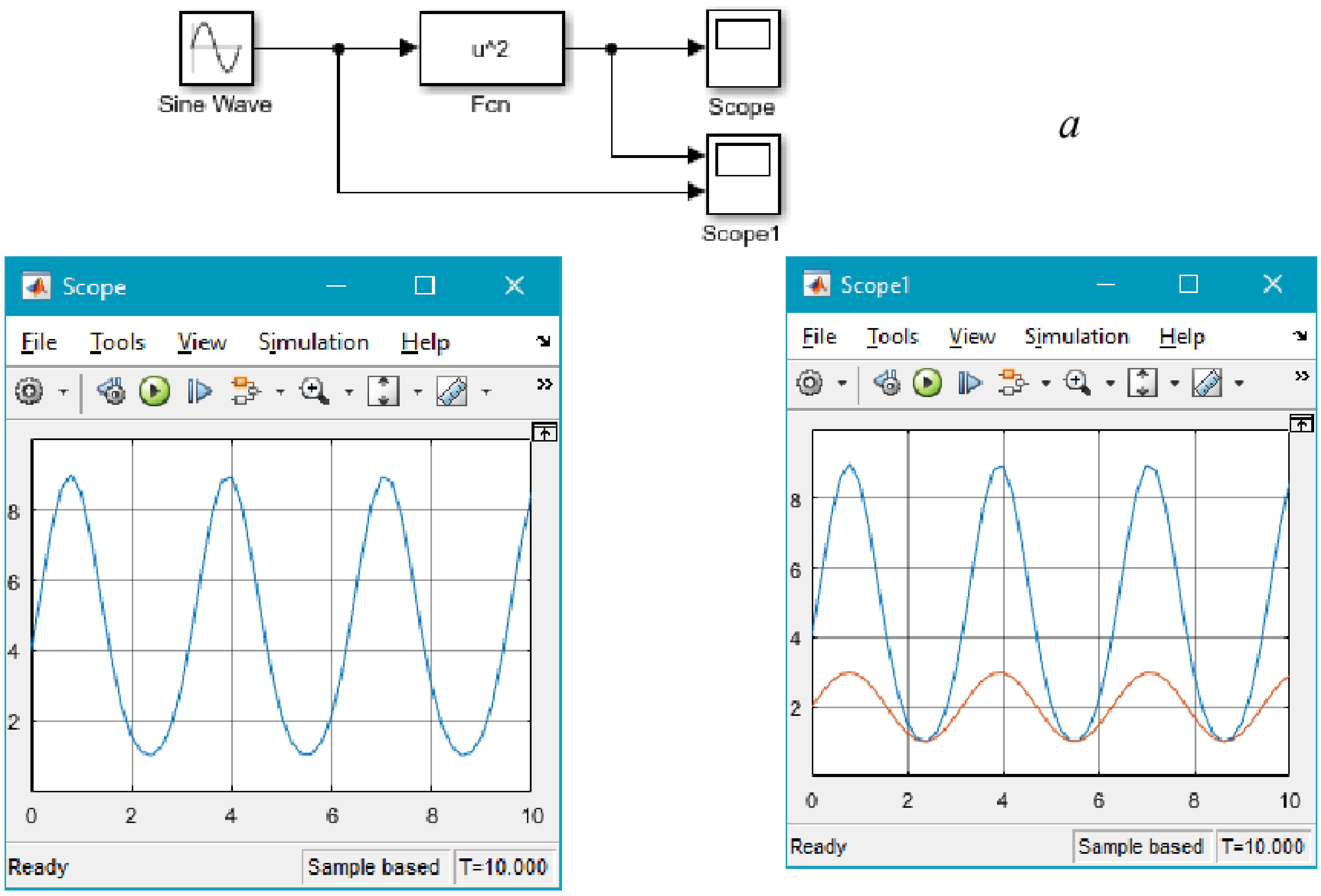
3. Если седловая точка отсутствует, то решение следует искать в смешанных стратегиях.

Пример3. Найти решение игры, заданной матрицей

$$A = \begin{vmatrix} 1 & 5 & 4 \\ 3 & 2 & 6 \\ 8 & 3 & 4 \end{vmatrix}$$

Решение. Исследование в матричных играх начинается с нахождения седловой точки в чистых стратегиях. Если матричная игра имеет седловую точку в чистых стратегиях, то нахождением этой точки заканчивается решение игры и записывается ответ: цена игры, соответствующая найденной седловой точке и стратегии первого и второго игроков соответствующие этой цене игры. Если же игра не имеет седловой точки в чистых стратегиях, то найденные нижняя и верхняя чистые цены этой игры указывают, что первый игрок не должен надеяться на выигрыш больший, чем верхняя цена игры, и может быть уверен в получении выигрыша не меньше нижней цены игры.

Решение начинаем с исследования платежной матрицы игры. Прежде всего, проверим наличие седловой точки в данной матрице *A*. Для этого найдем минимальные элементы в каждой из строк (1, 2 и 3) и максимальные элементы в каждом из столбцов (8, 5 и 6).



a

б *c*

Рисунок 1.9 – Структурная схема модели сигнала $x(t)$ =$(2+sin(2t))^2$– *a* и графики двух сигналов на экране виртуального осциллографа: 2+$sin(2t)$ –*б*, и $x(t)$ =2+sin(2t) – *c*

Задание 1.3. Моделирование интегрированного сигнала

В системах связи помехи и замирания, воздействующие на сигнал при его прохождении по каналу связи, имеют статистический характер и могут быть описаны при помощи различных законов распределений. В частности, замирания в канале связи при отсутствии прямой видимости между абонентом и базовой станцией имеют рэлеевский закон распределения; аддитивные помехи (шумы) часто описываются нормальным (гауссовским) законом распределения; временные интервалы между вызовами в сетях связи обычно имеют экспоненциальный закон распределения и.т.д.

Построить модель для имитации смеси сигналов:

- синусоидального сигнала и белого шума (рисунок 1.10);
- синусоидального сигнала и гауссова шума (с математическим ожиданием m=0,и дисперсией σ^2 =0.01);
- синусоидального сигнала и шума с равномерным распределением.

применяет свои чистые стратегии A$_1$, A$_2$ и A$_3$ с вероятностями p_1, p_2 и p_3 соответственно. Сумма вероятностей равна единице. Тогда, если второй игрок применит свои чистые стратегии B$_1$ или B$_2$ или B$_3$, то игрок A получит средний выигрыш, больше или равный цене игры, т. е. получим следующую систему неравенств:

$$\begin{cases} p_1+3p_2+8p_3 \geq v \ (\text{при стратегии } B_1) \\ 5p_1+2p_2+3p_3 \geq v \ (\text{при стратегии } B_2) \end{cases} \qquad (8.2)$$

Так как сумма вероятностей равна единице, получим следующее уравнение:

$$p_1+p_2+p_3=1 \qquad (8.3)$$

Разделим все члены неравенств системы (1.9) и уравнение (1.10) на v и заменим полученные выражения соответственно на: $y_1=\frac{p_1}{v}, y_2=\frac{p_2}{v}, y_3=\frac{p_3}{v}$ Система (4.9) и уравнение (4.10) примет следующий вид:

$$\begin{cases} y_1+3y_2+8y_3 \geq 1 \\ 5y_1+2y_2+3y_3 \geq 1 \\ 4y_1+6y_2+4y_3 \geq 1 \end{cases}$$

Так как первый игрок стремиться сделать свой выигрыш как можно больше, т. е. сделать значение цены игры максимальным, следовательно, сумма $y_1+y_2+y_3 \rightarrow \min$. Таким образом, получим следующую задачу линейного программирования:

Найти значения y_1, y_2, y_3 при которых целевая функция $\Psi=y_1+y_2+y_3$ примет минимальное значение:

$$\Psi=y_1+y_2+y_3 \rightarrow \min$$

при условии:

$$\begin{cases} y_1+3y_2+8y_3 \geq 1 \\ 5y_1+2y_2+3y_3 \geq 1 \end{cases} \qquad (8.4)$$

Найдя решения полученной задачи линейного программирования, вычислим цену игры и соответствующие вероятности по формулам:

$$v=\frac{1}{\Psi_{\min}}=\frac{1}{\sum_{i=1}^{m} y_i}\;;\qquad \frac{y_i}{v}=p_i\qquad (i=\overline{1,m}).$$

Проведя аналогичные рассуждения для второго игрока, получим задачу линейного программирования двойственную к задаче (4.11):

Найти значения x_1, x_2, x_3 при которых целевая функция $F=x_1+x_2+x_3$ примет максимальное значение:

$$F=x_1+x_2+x_3 \rightarrow \max$$

при условии:

$$x_1+5x_2+4x_3\le 1;\; 3x_1+2x_2+6x_3\le 1 \qquad (8.5)$$

$$\frac{x_j}{v}=q_j \qquad (j=\overline{1,n}). \qquad \text{или} \qquad v=\frac{1}{\sum_{j=1}^{n} x_j}$$

Пример решения задачи в Excel для второго игрока приведен на рисунке 8.2 (алгоритм решения приведен ниже).

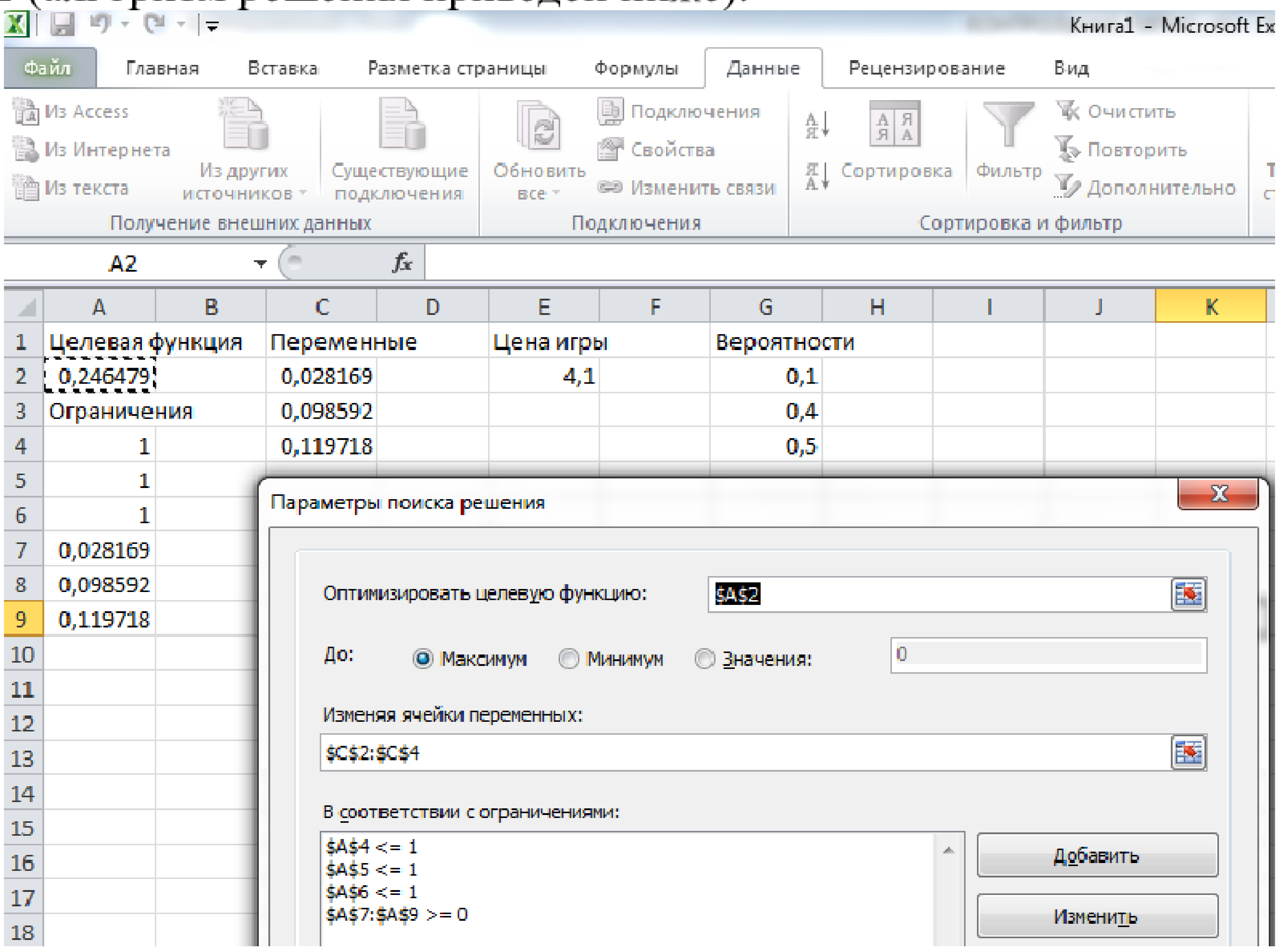


Рисунок 8.2 – Рабочий лист с результатами решения задачи поиска смешанных стратегий для второго игрока и заполненными полями окна диалога «Параметры поиска решения»

Аналогично находиться решение для первого игрока (рисунок 8.3).

Ответ: Оптимальная стратегия каждого игрока состоит в том, чтобы чередовать свои чистые стратегии случайным образом, т. е. первому игроку, чтобы выиграть не менее 4, надо чередовать применение первой и третьей стратегии, выбирая каждую с вероятностью 0,5, а вторую стратегию не применять. Второму игроку, чтобы не дать выиграть первому игроку больше, чем 4, необходимо в 4 раза реже применять свою первую чистую стратегию по сравнению со второй и в 5 раз реже по сравнению с третьей.

При решении задачи теории игр рекомендуется использовать следующий алгоритм:

1. Исключить из платежной матрицы игры заведомо невыгодные стратегии. Такими стратегиями для первого игрока (второго игрока) являются те, которым соответствуют строки (столбцы) с элементами, меньшими (большими) по сравнению с элементами других строк (столбцов).

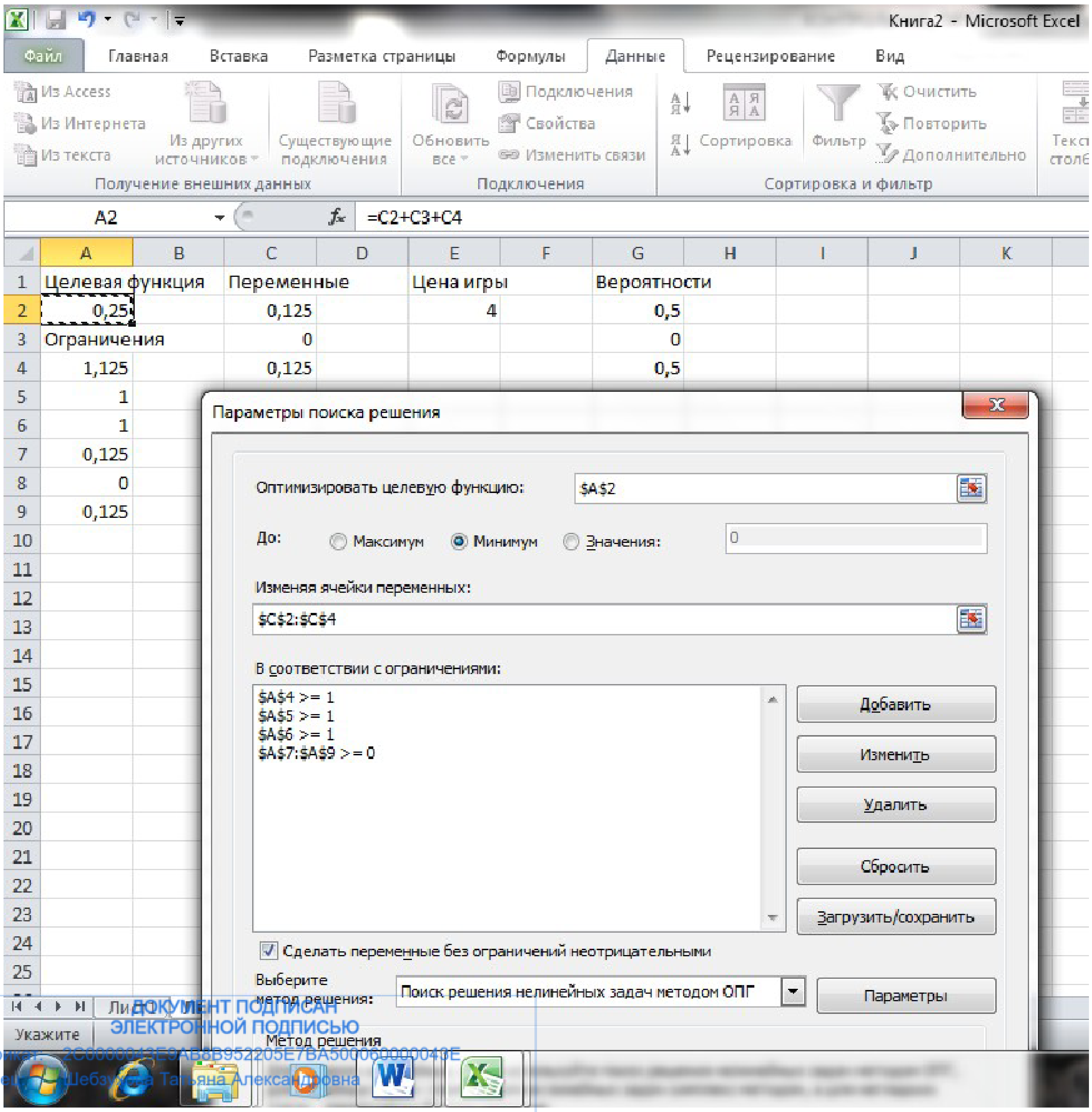


Рисунок 8.3 – Рабочий лист с результатами решения задачи для первого игрока и заполненными полями окна диалога «Параметры поиска решения»

2. Определить имеет ли игра седловую точку. Для этого необходимо найти верхнюю и нижнюю цену игры:

2.1. Если они равны, то игра имеет седловую точку, а значит решение в чистых стратегиях и сразу можно записать ответ, так как соответствующие ей стратегии игроков будут оптимальными.

2.2. Если верхняя и нижняя цены игры не равны, то необходимо искать решение игры в смешанных стратегиях путем сведения ее к задаче линейного программирования.

3. Процесс нахождения решения игры с использованием методов линейного программирования включает следующие этапы:

3.1. Составляют пару двойственных задач линейного программирования, эквивалентных данной матричной игре.

3.2. Определяют оптимальные планы пары двойственных задач.

3.3. Используя соотношение между планами пары двойственных задач, оптимальными стратегиями и ценой игры, находят решение игры.

Пример 3. Найти решение игры, определяемой матрицей:

$$A=\begin{pmatrix} 0 & 1 & -1 \\ 0 & -1 & 0 \\ 1 & 0 & -1 \end{pmatrix}.$$

Решение. При решении этой игры к каждому элементу матрицы A прибавим 1 и получим следующую матрицу:

$$A_1=\begin{pmatrix} 1 & 2 & 0 \\ 1 & 0 & 1 \\ 2 & 1 & 0 \end{pmatrix}.$$

Так как игра не имеет седловой точки, составим теперь пару взаимно-двойственных задач:

$$y_1+y_2+y_3 \to \min \qquad\qquad x_1+x_2+x_3 \to \max$$

$$\begin{cases} y_1+y_2+2y_3 \ge 1, \\ 2y_1+\qquad\quad y_3 \ge 1, \\ \qquad\qquad\quad y_3 \ge 1, \\ y_1, y_2, y_3 \ge 0. \end{cases} \qquad\qquad \begin{cases} x_1+2x_2 \qquad \le 1, \\ x_1 \qquad\quad +x_3 \le 1, \\ 2x_1+x_2 \qquad \le 1, \\ x_1, x_2, x_3 \ge 0. \end{cases}$$

Рассмотрим пример решения задачи с использованием инструмента «Поиск решения» электронной таблицы Excel.

Для определения оптимального плана второй задачи необходимо:

1. Записать целевую функцию в следующем виде:

$$f'(c_1, c_2, c_3)=c_1+c_2+c_3 \quad ,$$

а ограничения – неравенства, следующим образом:

$$c_1+2*c_2\le 1$$
$$c_1+c_3\le 1$$
$$2*c_1+c_2\le 1$$
$$c_1\ge 0, c_2\ge 0, c_3\ge 0.$$

2. Выполнить следующие действия:

а) ввести в ячейку A_1 выражение для целевой функции: $=c_1+c_2+c_3$;

б) ввести ограничения в ячейки $A_3 : A_9$ и нулевые значения в ячейки $C_1 : C_3$;

в) выполнить команду **Сервис/Поиск решения** и заполнить параметры в окне диалога **Поиск решений**. Установить целевую ячейку А1, равной максимальному значению, в поле ввода **Изменяя ячейки:** ввести: c1:c3. Перейти в поле ввода **Ограничения:** и выполнить команду **Добавить**. Заполнить появившееся окно диалога, последовательно вводя ячейки, в которых находятся неравенства системы ограничений;

г) решить задачу, выбрав команду **Выполнить**, в появившемся окне **Результаты поиска решения** выбрать тип отчета **Результаты**;

д) вычислить значение цены игры. Для этого в ячейку e2 ввести формулу: =1/A1. Цена игры равна 0,67;

е) рассчитать значения вероятностей использования игроком своих стратегий. Для этого ввести в ячейки E4:E6 соответственно: =c1·e2; =c2·e2; =c3·e2. Вероятность использования игроком первой стратегии равна 0, второй стратегии – 0,33, третьей – 0,67.

Аналогично решая задачу, находим вероятность использования своих стратегий вторым игроком. Вероятность использования игроком первой стратегии равна 0,33, второй стратегии – 0,67, третьей – 0.

Пример 4. Построение матричной игры на основе симметричной пары двойственных задач

Для всякой матричной игры можно записать симметричную пару двойственных задач. Справедливо и обратное: для всякой симметричной пары двойственных задач можно записать матричную игру.

Пусть задана симметричная пара двойственных задач: прямая задача:

$F=CX$, $AX \le B$, $X \ge 0$, двойственная задача: $\Psi= BY$, $YA\ge C, Y\ge 0$. Тогда этой симметричной паре двойственных задач можно поставить в соответствие игру, определяемую матрицей

$$D=\begin{bmatrix} 0 & A & -B \\ -A^T & 0 & C^T \\ B^T & -C & 0 \end{bmatrix}, \qquad (8.6)$$

где индекс ^{T} означает операцию транспонирования.

Следует отметить, что если каждая матричная игра имеет оптимальные стратегии, то не всякая задача линейного программирования имеет решение.

Построить игру, определяемую следующей парой двойственных задач:

прямая задача:

$$F = 2x_1 + 3x_2 (\max)$$

$$при\ ограничениях ¿$$

$$\{2x_1 + x_2 \le 10 ¿ \{-x_1 + 3x_2 \le 12 ¿¿¿$$

$$¿ \qquad (8.7)$$

двойственная задача:

$$F = 10y_1 + 12y_2 (\min),$$

$$при\ ограничениях ¿$$

$$\{2y_1 - y_2 \ge 2 ¿ \{y_1 + 3y_2 \ge 3 ¿¿¿$$

$$¿ \qquad (8.8)$$

Решение. Запишем матрицы, элементами которых являются: коэффициенты при неизвестных левой части системы ограничений прямой и двойственной задач соответственно (матрицы A и A^{T}); коэффициенты при неизвестных в целевой функции прямой и двойственной задач соответственно (матрицы C и B^{T}); правые части системы ограничений прямой и двойственной задач соответственно (матрицы B и C^{T}):

$$A = \begin{bmatrix} 2 & 1 \\ -1 & 3 \end{bmatrix};\ A^T = \begin{bmatrix} 2 & -1 \\ 1 & 3 \end{bmatrix};\ B = \begin{bmatrix} 10 \\ 12 \end{bmatrix}.$$

$$B^T = (10, 12);\ C = (2, 3);\ C^T = \begin{bmatrix} 2 \\ 3 \end{bmatrix}.$$

Следовательно, исходной симметричной паре двойственных задач можно поставить в соответствие матричную игру, определяемую матрицей:

$$D = \begin{bmatrix} 0 & 0 & 2 & 1 & -10 \\ 0 & 0 & -1 & 3 & -12 \\ -2 & 1 & 0 & 0 & 2 \\ -1 & -3 & 0 & 0 & 3 \\ 10 & 12 & -2 & -3 & 0 \end{bmatrix}.$$

Аппаратура и материалы. Для выполнения лабораторной работы необходимо использовать следующее: аппаратное обеспечение: персональный компьютер и программное обеспечение: операционную систему Windows 10 и выше; Microsoft Office13 и выше, системы компьютерной математики Machcad и MATLAB R2017a и выше.

Указания по технике безопасности. Самостоятельно не производить установку и удаление программного обеспечения; ремонт персонального компьютера. Соблюдать правила технической эксплуатации и техники безопасности при работе с электрооборудованием.

Методика и порядок выполнения работы

Выполните предложенные задания, предварительно разобрав примеры, приведенные выше.

Задание 8.1. Принятие решений в условиях неопределенности

1. Исследуйте платежную матрицу игры (таблица 8.1):
- определите нижнюю и верхнюю цены игры;
- определите минимаксные и максиминные стратегии;
- определите оптимальные решения игры в чистых стратегиях, если существует седловая точка.

2. Найдите решения игры, не имеющей седловую точку, путем сведения ее к задаче линейного программирования:
- запишите модели прямой и двойственной задачи;
- найдите решение обеих задач;
- проанализируйте правильность найденного решения с помощью двух теорем двойственности.

3. Сделайте вывод.

Таблица 8.1– Варианты второго задания

№ варианта	Платежная матрица игры	№ варианта	Платежная матрица игры
1.	$\left\|\begin{matrix} 2 & 5 & 8 \\ 7 & 2 & 8 \\ 3 & 7 & 1 \end{matrix}\right\|$	1.	$\left\|\begin{matrix} 4 & 5 & 3 \\ 6 & 8 & 4 \\ 4 & 2 & 4 \end{matrix}\right\|$
	$\left\|\begin{matrix} 2 & 5 & 8 \\ 7 & 2 & 8 \\ 9 & 7 & 7 \end{matrix}\right\|$		$\left\|\begin{matrix} 4 & 5 & 8 \\ 6 & 9 & 5 \\ 5 & 2 & 1 \end{matrix}\right\|$
2.	$\left\|\begin{matrix} 3 & 5 & 6 \\ 5 & 3 & 8 \\ 3 & 7 & 1 \end{matrix}\right\|$	2.	$\left\|\begin{matrix} 2 & 5 & 8 \\ 7 & 3 & 9 \\ 3 & 5 & 1 \end{matrix}\right\|$
	$\left\|\begin{matrix} 3 & 3 & 8 \\ 7 & 1 & 8 \\ 3 & 3 & 2 \end{matrix}\right\|$		$\left\|\begin{matrix} 3 & 4 & 9 \\ 3 & 2 & 8 \\ 1 & 7 & 1 \end{matrix}\right\|$

3.	$\begin{vmatrix} 2 & 5 & 9 \\ 7 & 2 & 7 \\ 3 & 3 & 1 \end{vmatrix}$	3.	$\begin{vmatrix} 6 & 5 & 8 \\ 7 & 1 & 8 \\ 3 & 4 & 1 \end{vmatrix}$
	$\begin{vmatrix} 3 & 5 & 4 \\ 7 & 2 & 3 \\ 5 & 7 & 5 \end{vmatrix}$		$\begin{vmatrix} 4 & 1 & 7 \\ 7 & 2 & 8 \\ 3 & 6 & 2 \end{vmatrix}$
4.	$\begin{vmatrix} 2 & 5 & 8 \\ 7 & 3 & 5 \\ 3 & 7 & 7 \end{vmatrix}$	4.	$\begin{vmatrix} 6 & 5 & 3 \\ 2 & 3 & 7 \\ 3 & 7 & 1 \end{vmatrix}$
	$\begin{vmatrix} 2 & 5 & 8 \\ 4 & 3 & 5 \\ 5 & 7 & 7 \end{vmatrix}$		$\begin{vmatrix} 5 & 3 & 8 \\ 6 & 2 & 7 \\ 3 & 3 & 1 \end{vmatrix}$
5.	$\begin{vmatrix} 5 & 5 & 8 \\ 6 & 2 & 7 \\ 3 & 4 & 1 \end{vmatrix}$	5.	$\begin{vmatrix} 2 & 4 & 9 \\ 4 & 3 & 5 \\ 5 & 8 & 7 \end{vmatrix}$
	$\begin{vmatrix} 5 & 8 & 3 \\ 3 & 2 & 7 \\ 3 & 7 & 1 \end{vmatrix}$		$\begin{vmatrix} 2 & 5 & 8 \\ 4 & 3 & 5 \\ 9 & 7 & 3 \end{vmatrix}$
6.	$\begin{vmatrix} 9 & 2 & 8 \\ 3 & 2 & 4 \\ 3 & 7 & 1 \end{vmatrix}$	6.	$\begin{vmatrix} 2 & 5 & 8 \\ 7 & 3 & 5 \\ 5 & 6 & 4 \end{vmatrix}$
	$\begin{vmatrix} 6 & 5 & 8 \\ 7 & 2 & 8 \\ 3 & 5 & 1 \end{vmatrix}$		$\begin{vmatrix} 2 & 6 & 9 \\ 7 & 3 & 5 \\ 6 & 6 & 9 \end{vmatrix}$
7.	$\begin{vmatrix} 3 & 5 & 9 \\ 7 & 4 & 6 \\ 3 & 7 & 2 \end{vmatrix}$	7.	$\begin{vmatrix} 2 & 3 & 8 \\ 4 & 4 & 5 \\ 5 & 7 & 7 \end{vmatrix}$
	$\begin{vmatrix} 2 & 5 & 8 \\ 7 & 2 & 8 \\ 3 & 7 & 1 \end{vmatrix}$		$\begin{vmatrix} 2 & 5 & 8 \\ 4 & 3 & 5 \\ 5 & 6 & 9 \end{vmatrix}$
8.	$\begin{vmatrix} 1 & 5 & 8 \\ 7 & 4 & 3 \\ 3 & 7 & 1 \end{vmatrix}$	8.	$\begin{vmatrix} 3 & 5 & 8 \\ 2 & 2 & 5 \\ 1 & 7 & 1 \end{vmatrix}$
	$\begin{vmatrix} 4 & 5 & 4 \\ 7 & 4 & 3 \\ 3 & 7 & 1 \end{vmatrix}$		$\begin{vmatrix} 7 & 5 & 8 \\ 4 & 3 & 5 \\ 5 & 7 & 7 \end{vmatrix}$

Задание 8.2. Постройте для симметричных пар двойственных задач определяемые ими матричные игры:

$$F=3x_1+4x_2+x_3(\max); \qquad F^{*}=12y_1+14y_2+16y_3(\max);$$

$$\begin{cases} 2x_1+3x_2-4x_3\le 12 \\ -x_1+x_2+x_3\le 14 \\ 3x_1+4x_2-x_3\le 16 \end{cases} \qquad \begin{cases} 2y_1-y_2+3y_3\ge 3 \\ 3y_1+y_2+4y_3\ge 4 \\ -4y_1+y_2-y_3\ge 1 \end{cases}$$

$$F=5x_1+7x_2+8x_3(\max); \qquad F^{*}=18y_1+16y_2+24y_3(mim);$$

$$\begin{cases} x_1+x_2+x_3\le 18 \\ 3x_1+2x_2+x_3\le 16 \\ 4x_1+3x_2+x_3\le 24 \end{cases} \qquad \begin{cases} y_1+3y_2+4y_3\ge 5 \\ y_1+2y_2+3y_3\ge 7 \\ 5y_1+y_2+y_3\ge 8 \end{cases}$$

Для исходной задачи составьте двойственную и постройте для полученной пары двойственных задач определяемые ими матричные игры:

$$F=2x_1+x_2+5x_3(\max); \qquad F=x_1+2x_2+3x_3(\max);$$

$$\begin{cases} x_1-x_2-x_3\le 4 \\ x_1-x_2+x_3\ge 5 \\ 2x_1-x_2+3x_3\ge 6 \end{cases} \qquad \begin{cases} 2x_1+2x_2-x_3\ge 2 \\ x_1-2x_2-4x_3\le -3 \\ x_1+x_2-2x_3\ge 6 \end{cases}$$

Содержание отчета и его форма

Подготовьте отчет, в котором опишите технологию решения задач линейного программирования средствами Excel, используя задания своего варианта. Отчет по лабораторной работе должен содержать:

1) название работы;
2) цель лабораторной работы;
3) условия выполненных заданий и их решение;
4) выводы;
5) ответы на контрольные вопросы.

Вопросы для защиты работы:

1. Что такое теория игр?

2. Что понимают под конфликтной ситуацией?
3. Что лежит в основе теории игр?
4. Перечислите основные понятия теории игр и дайте им определение.
5. По каким признакам проводят классификацию игр?
6. Назовите известные Вам классификации.
7. Какие основные способы описания и анализа игры Вы знаете?
8. Что такое платежная матрица игры?
9. Как определить по платежной матрице нижнюю и верхнюю цену игры?
10. С чего начинают решение игры?
11. В каком случае игра имеет решение в чистых стратегиях?
12. Что такое смешанная стратегия?
13. В чем заключается поиск оптимальной смешанной стратегии?

ЛАБОРАТОРНАЯ РАБОТА 9
Случайные процессы с дискретным состоянием. Уравнения Колмогорова для стационарного режима

Цель и содержание: Научиться составлять уравнения Колмогорова и выполнять анализ работы моделируемой системы.

Организационная форма занятий: решение проблемных задач, разбор конкретных ситуаций

Вопросы для обсуждения на лабораторном занятии: построение, решение и анализ моделей систем, описываемых уравнениями Колмогорова.

Формируемые компетенции: ОПК-7.1 ОПК-7.2 ОПК-8.1 ОПК-8.2

Теоретическое обоснование

Изучите теоретический материал по данной теме.

Понятие случайного процесса

Строго говоря, случайные возмущения присущи любому процессу. Проще привести примеры случайного, чем «неслучайного» процесса. Даже, например, процесс хода часов (вроде бы это строгая выверенная работа – «работает как часы») подвержен случайным изменениям (уход вперед, отставание, остановка). Но до тех пор, пока эти возмущения несущественны, мало влияют на интересующие нас параметры, мы можем ими пренебречь и рассматривать процесс как детерминированный, неслучайный.

Пусть имеется некоторая система S (техническое устройство, группа таких устройств, технологическая система – станок, участок, цех, предприятие, отрасль промышленности и т.д.). В системе S протекает **случайный процесс**, если она с течением времени меняет свое состояние (переходит из одного состояния в другое), причем, заранее неизвестным случайным образом.

Примеры: 1. Система S – технологическая система (участок станков). Станки время от времени выходят из строя и ремонтируются. Процесс, протекающий в этой системе, случаен.

2. Система S – самолет, совершающий рейс на заданной высоте по определенному маршруту. Возмущающие факторы – метеоусловия, ошибки экипажа и т.д., последствия – «болтанка», нарушение графика полетов и т.д.

Марковский случайный процесс

Случайный процесс, протекающий в системе, называется **Марковским**, если для любого момента времени t_0 вероятностные характеристики процесса в будущем зависят только от его состояния в данный момент t_0 и не зависят от того, когда и как система пришла в это состояние.

Пусть в настоящий момент t_0 система находится в определенном состоянии S_0. Мы знаем характеристики состояния системы в настоящем $t_0 \to S_0$ и все, что было при $t < t_0$ (предысторию процесса). Можем ли мы предугадать (предсказать) будущее, т.е. что будет при $t > t_0$? В точности – нет, но какие-то вероятностные характеристики процесса в будущем найти можно. Например, вероятность того, что через некоторое время τ система S окажется в состоянии S_1 или останется в состоянии S_0 и т.д.

Пример. Система S – группа самолетов, участвующих в воздушном бою. Пусть x – количество «красных» самолетов, y – количество «синих» самолетов. К моменту времени t_0 количество сохранившихся (не сбитых) самолетов соответственно – x_0, y_0. Нас интересует вероятность того, что в момент времени $(t_0 + \tau)$ численный перевес будет на стороне «красных». Эта вероятность зависит от того, в каком состоянии находилась система в момент времени t_0, а не от того, когда и в какой последовательности погибали сбитые до момента t_0 самолеты.

На практике Марковские процессы в чистом виде обычно не встречаются. Но имеются процессы, для которых влиянием «предистории» можно пренебречь. И при изучении таких процессов можно применять Марковские модели (в теории массового обслуживания рассматриваются и не Марковские системы массового обслуживания, но математический аппарат, их описывающий, гораздо сложнее).

В исследовании операций большое значение имеют Марковские случайные процессы с дискретными состояниями и непрерывным временем.

Процесс называется **процессом с дискретным состоянием**, если его возможные состояния S_1, S_2, … можно заранее определить, и переход системы из состояния в состояние происходит «скачком», практически мгновенно.

Процесс называется **процессом с непрерывным временем**, если моменты возможных переходов из состояния в состояние не фиксированы заранее, а неопределенны, случайны и могут произойти в любой момент.

Далее рассматриваются только процессы с дискретным состоянием и непрерывным временем.

Пример1. Технологическая система (участок) S состоит из двух станков, каждый из которых в случайный момент времени может выйти из строя (отказать), после чего мгновенно начинается ремонт узла, тоже продолжающийся заранее неизвестное, случайное время. Возможны следующие состояния системы:

S_0 - оба станка исправны;

S_1 - первый станок ремонтируется, второй исправен;

S_2 - второй станок ремонтируется, первый исправен;

S_3 - оба станка ремонтируются.

Переходы системы S из состояния в состояние происходят практически мгновенно, в случайные моменты выхода из строя того или иного станка или окончания ремонта.

При анализе случайных процессов с дискретными состояниями удобно пользоваться геометрической схемой – **графом состояний**. Вершины графа – состояния системы. Дуги графа – возможные переходы из состояния в состояние. Для нашего примера граф состояний приведен на рисунке 9.1.

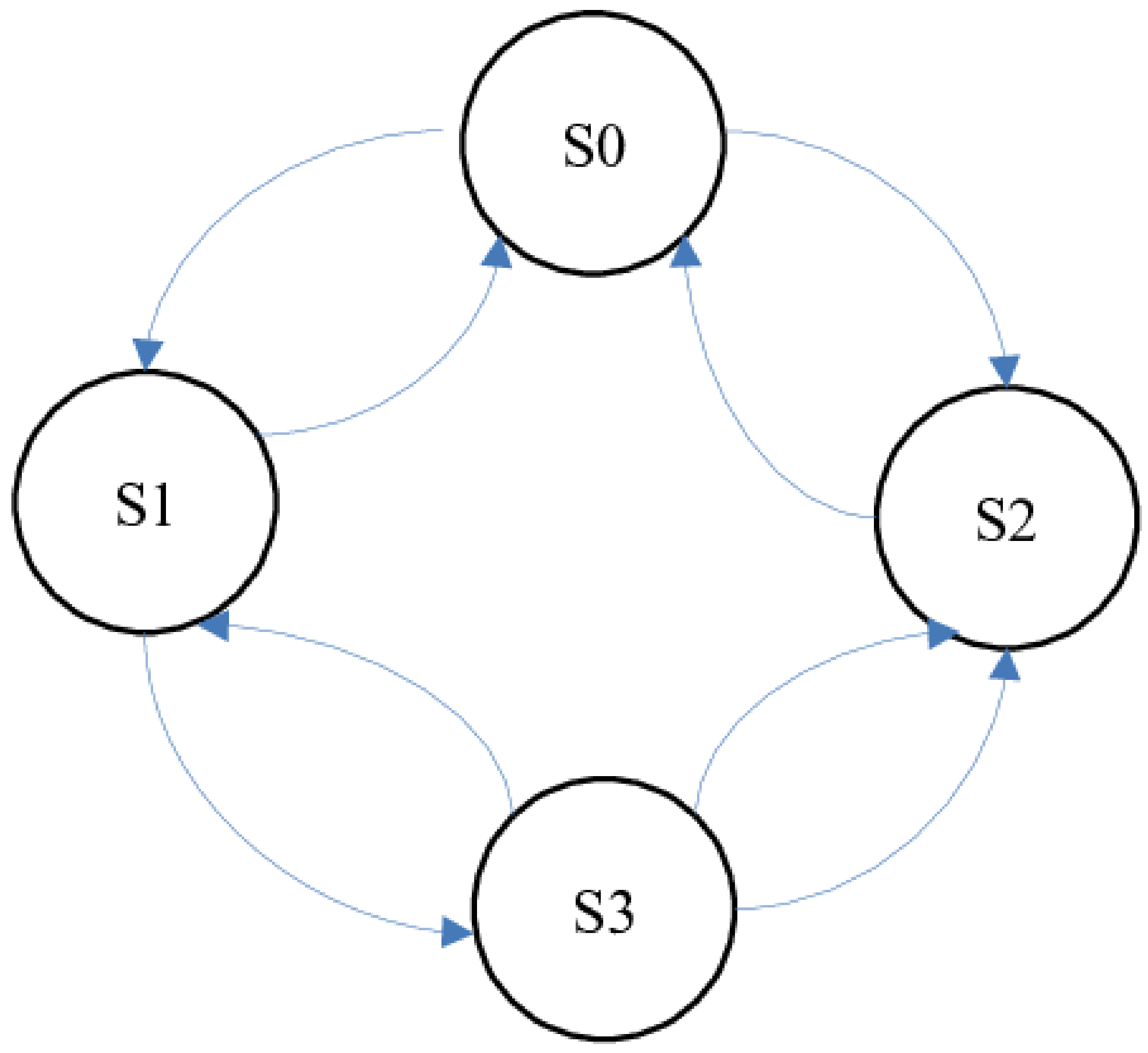


Рисунок 9.1 – Граф состояний системы

Примечание. Переход из состояния S_0 в S_3 на рисунке не обозначен, т.к. предполагается, что станки выходят из строя независимо друг от друга. Вероятностью одновременного выхода из строя обоих станков мы пренебрегаем.

Потоки событий

Поток событий – последовательность однородных событий, следующих одно за другим в какие-то случайные моменты времени.

В предыдущем примере – это поток отказов и поток восстановлений. Другие примеры: поток вызовов на телефонной станции, поток покупателей в магазине и т.д.

Поток событий можно наглядно изобразить рядом точек на оси времени $O\ t$ – рисунок 9.2.

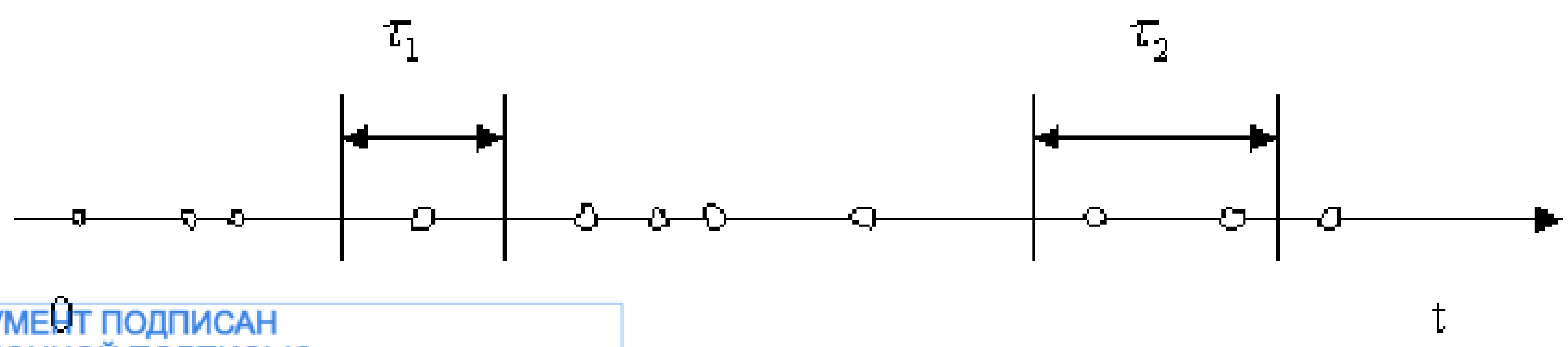


Рисунок 9.2. – Изображение потока событий на оси времени

Положение каждой точки случайно, и здесь изображена лишь какая-то одна реализация потока.

Интенсивность потока событий (λ) – это среднее число событий, приходящееся на единицу времени.

Рассмотрим некоторые свойства (виды) потоков событий.

Поток событий называется **стационарным**, если его вероятностные характеристики не зависят от времени.

В частности, интенсивность λ стационарного потока постоянна. Поток событий неизбежно имеет сгущения или разрежения, но они не носят закономерного характера, и среднее число событий, приходящееся на единицу времени, постоянно и от времени не зависит.

Поток событий называется **потоком без последствий**, если для любых двух непересекающихся участков времени τ_1 и τ_2 (см. рисунок 9.2.) число событий, попадающих на один из них, не зависит от того, сколько событий попало на другой. Другими словами, это означает, что события, образующие поток, появляются в те или иные моменты времени **независимо друг от друга** и вызваны каждое своими собственными причинами.

Поток событий называется **ординарным**, если события в нем появляются поодиночке, а не группами по нескольку сразу.

Поток событий называется **простейшим (или стационарным пуассоновским),** если он обладает сразу тремя свойствами:

1) стационарен,

2) ординарен,

3) не имеет последствий.

Простейший поток имеет наиболее простое математическое описание. Он играет среди потоков такую же особую роль, как и закон нормального распределения среди других законов распределения. А именно, при наложении достаточно большого числа независимых, стационарных и ординарных потоков (сравнимых между собой по интенсивности) получается поток, близкий к простейшему.

Для простейшего потока с интенсивностью λ интервал *T* между соседними событиями имеет так называемое **показательное (экспоненциальное) распределение** с плотностью

$$f(t) = \lambda e^{-\lambda t},$$

где λ - параметр показательного закона.

Для случайной величины *T*, имеющей показательное распределение, математическое ожидание m_T есть величина, обратная параметру, а среднее квадратичное отклонение σ_T равно математическому ожиданию

$$m_T = \sigma_T = 1/\lambda.$$

Уравнения Колмогорова для вероятностей состояний. Финальные вероятности состояний

Рассматривая Марковские процессы с дискретными состояниями и непрерывным временем, подразумевается, что все переходы системы *S* из состояния в состояние происходят под действием простейших потоков событий (потоков вызовов, потоков отказов, потоков восстановлений и т.д.). Если все потоки событий, переводящие систему *S* из состояния в состояние простейшие, то процесс, протекающий в системе, будет Марковским.

Итак, на систему, находящуюся в состоянии S_i, действует простейший поток событий. Как только появится первое событие этого потока, происходит «перескок» системы из состояния S_i в состояние S_j (на графе состояний по стрелке $S_i \to S_j$).

Для наглядности на графе состояний системы у каждой дуги проставляют интенсивности того потока событий, который переводит систему по данной дуге (стрелке). λ_{ij} - интенсивность потока событий, переводящий систему из состояния S_i в S_j. Такой граф называется **размеченным**. Для нашего примера размеченный граф приведен на рисунок 9.3.

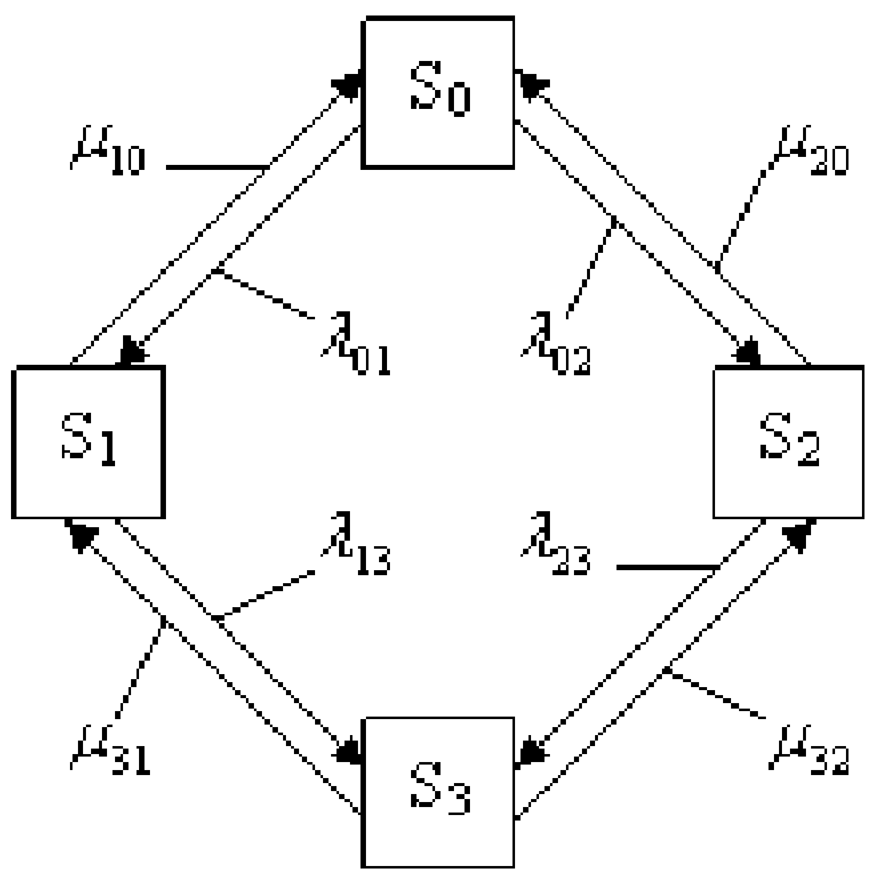


Рисунок 9.3. – Размеченный граф состояний системы

На этом рисунке λ_{ij} - интенсивности потока отказов; μ_{ij} - интенсивности потока восстановлений.

Предполагаем, что среднее время ремонта станка не зависит от того, ремонтируется ли один станок или оба сразу. Т.е. ремонтом каждого станка занят отдельный специалист.

Пусть система находится в состоянии S_0. В состояние S_1 ее переводит поток отказов первого станка. Его интенсивность равна

$$\lambda_{01} = 1 / T_{ср.раб.1}, ед.времени^{-1},$$

где $T_{ср.раб.1}$ – среднее время безотказной работы первого станка.

Из состояния S_1 в S_0 систему переводит поток «окончаний ремонтов» первого станка. Его интенсивность равна

$$\mu_{10} = 1 / T_{ср.рем.1}, ед.времени^{-1},$$

где $T_{ср.рем1}$ - среднее время ремонта первого станка.

Аналогично вычисляются интенсивности потоков событий, переводящих систему по всем дугам графа. Имея в своем распоряжении размеченный граф состояний системы, строится **математическая модель** данного процесса.

Пусть рассматриваемая система *S* имеет n - возможных состояний $S_1, S_2, ..., S_n$. Вероятность i - го состояния $p_i(t)$ - это вероятность того, что в момент времени t система будет находиться в состоянии S_i. Очевидно, что для любого момента времени сумма всех вероятностей состояний равна единице:

$$\sum_{i=1}^{n} p_i(t) = 1.$$

Для нахождения всех вероятностей состояний $p_i(t)$ как функций времени составляются и решаются **уравнения Колмогорова** – особого вида уравнения, в которых неизвестными функциями являются вероятности состояний. Правило составления этих уравнений приведем здесь без доказательств. Но прежде, чем его приводить, объясним понятие **финальной вероятности состояния**.

Что будет происходить с вероятностями состояний при $t \to \infty$? Будут ли $p_1(t), p_2(t),...$ стремиться к каким-либо пределам? Если эти пределы существуют и не зависят от начального состояния системы, то они называются **финальными вероятностями состояний**.

$$\lim_{t \to \infty} p_i(t) = p_i, i = \overline{1,n},$$

где n - конечное число состояний системы.

Финальные вероятности состояний – это уже не переменные величины (функции времени), а постоянные числа. Очевидно, что

$$\sum_{i=1}^{n} p_i = 1.$$

Финальная вероятность состояния S_i – это по – существу среднее относительное время пребывания системы в этом состоянии.

Например, система *S* имеет три состояния S_1, S_2 и S_3. Их финальные вероятности равны соответственно 0,2; 0,3 и 0,5. Это значит, что система в предельном стационарном состоянии в среднем 2/10 времени проводит в состоянии S_1, 3/10 – в состоянии S_2 и 5/10 – в состоянии S_3.

Правило составления системы уравнений Колмогорова: в каждом уравнении системы **в левой его части** стоит финальная вероятность данного состояния p_i, умноженная на суммарную интенсивность всех потоков, **ведущих из данного состояния**, а **в правой его части** сумма произведений интенсивностей всех потоков,