

ГУАП

КАФЕДРА № 14

ОТЧЕТ
ЗАЩИЩЕН С ОЦЕНКОЙ
ПРЕПОДАВАТЕЛЬ

должность, уч. степень, звание

подпись, дата

инициалы, фамилия

ОТЧЕТ О ЛАБОРАТОРНОЙ РАБОТЕ №3
Решение задач линейного программирования

по курсу:

Теория принятия решений

РАБОТУ ВЫПОЛНИЛ

СТУДЕНТ гр. №

1145

24.11.2023  Алейник С П

подпись, дата

инициалы, фамилия

Санкт-Петербург 2023

Оглавление

ПОСТАНОВКА ЗАДАЧИ2

МАТЕМАТИЧЕСКАЯ МОДЕЛЬ2

 УРАВНЕНИЯ2

 МАТРИЦЫ2

ПРОГРАММА2

 СЦЕНАРИЙ4

РЕЗУЛЬТАТЫ МОДЕЛИРОВАНИЯ5

 ГРАФИКИ6

ВЫВОДЫ6

Постановка задачи

Целью работы является исследование метода решения задач линейного программирования, построив алгоритм и отладив программу реализации для применения функции *linprog* математического пакета *MatLab*.

Вариант	Порядок задачи линейного программирования	Число неравенств	Число равенств
8	7	9	3

Математическая модель

В рамках данной математической модели рассматривается задача линейного программирования общего вида.

Уравнения

$$c^T \cdot x \rightarrow \min$$

$$A_1 \cdot x \leq b_1$$

$$A_2 \cdot x = b_2$$

$$x^m \leq x \leq x^M$$

Матрицы

Нет

Программа

```
NOR=100;
    hide_ind_results= true;

    avg_iterations = 0;
    avg_time = 0;

    for run = 1:NOR
        found_solution = false;
        iterations = 0;
        tic;
```

```

while ~found_solution
    iterations = iterations + 1;
    matrixA1 = rand(9, 7); % Неравенства
    matrixA2 = rand(3, 7); % Равенства

    xM = rand(7, 1) * 5;
    xm = zeros(7, 1);

    b1 = rand(9, 1);
    b2 = rand(3, 1);
    c = ones(7, 1);

    options = optimoptions('linprog', 'Display', 'off');
    [x, C] = linprog(c, matrixA1, b1, matrixA2, b2, xm, xM,
options);

    if ~isempty(x)
        found_solution = true;
    end
end

time_elapsed = toc;
avg_iterations = avg_iterations + iterations;
avg_time = avg_time + time_elapsed;
if ~hide_ind_results
    disp("X vector:")
    disp(x);
    disp("C value: ")
    disp(C);
    disp(['Time taken: ' num2str(time_elapsed) ' seconds']);
    disp(['Number of iterations: ' num2str(iterations)]);
end
end

avg_iterations = avg_iterations / NOR;
avg_time = avg_time / NOR;

disp(' ');
disp(['Number of runs: ' num2str(NOR)])
disp(['Average number of iterations: ' num2str(avg_iterations)]);
disp(['Average time taken: ' num2str(avg_time) ' seconds']);
break;

```

Листинг 1 - Программа

Сценарий

X vector:

0.1141
0
0
0.0885
0
0
0.0441

C value:

0.2468

matrixA1

0.3795	0.9866	0.6428	0.2723	0.7445	0.2193	0.7269
0.9554	0.6719	0.5216	0.2906	0.1147	0.8643	0.0483
0.9545	0.2485	0.8877	0.9616	0.7119	0.2791	0.4877
0.3634	0.6579	0.8704	0.1439	0.8341	0.3077	0.7823
0.6975	0.5341	0.0062	0.6833	0.1026	0.0069	0.8704
0.4592	0.4725	0.8463	0.4865	0.1393	0.5087	0.3830
0.5372	0.1927	0.8640	0.9699	0.0518	0.4980	0.7504
0.4826	0.7300	0.3326	0.7878	0.9378	0.9011	0.1254
0.8513	0.2063	0.5540	0.8145	0.9399	0.0102	0.5465

matrixA2

0.7815	0.5864	0.3785	0.9663	0.2819	0.3099	0.0453
0.6380	0.8840	0.8289	0.8439	0.2604	0.8321	0.9908
0.9185	0.0354	0.0038	0.4450	0.4119	0.1688	0.1309

xM

3.9179
4.4843
2.4057
2.5075
0.9570
1.4475
3.2568

xm

0
0
0
0
0
0
0
0

c

1
1
1
1
1
1
1
1

Time taken: 2.3449 seconds

Number of iterations: 350

Number of runs: 1

Average number of iterations: 350

Average time taken: 2.3449 seconds

>>

Функции

Нет

Результаты моделирования

Результатом моделирования является код, который решает задачу линейного программирования, где случайным образом генерируются коэффициенты матриц и векторов для неравенств, равенств, а также вектор функции для оптимизации. Затем используется функция `linprog` из MATLAB для решения этой задачи. Результатом функции `linprog` является вектор x , который является оптимальным решением задачи оптимизации, и значение C , которое представляет оптимальное значение целевой функции.

Средние значения за 100 проходов:

- Среднее время выполнения программы: 622ms
- Среднее количество итераций: 134

Диалог

Lab3

ЛР3

>> lab3

ЛР3

...

X vector:

0
0.0305
0.0759

0
0.3712
0.0981
0.1359

C value:
0.7116

Time taken: 0.21632 seconds
Number of iterations: 53

Number of runs:5
Average number of iterations: 97
Average time taken: 0.39802 seconds

Графики

Нет

Выводы

В ходе работы был изучен и протестирован метод решения линейных задач оптимизации с помощью функции `linprog` в MATLAB. Разработанный алгоритм успешно решает задачи с произвольными параметрами, генерируемыми случайным образом. Полученные оптимальные значения переменных и целевой функции подтверждают эффективность метода.