Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Московский государственный технический университет
имени Н.Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н.Э. Баумана)

ФАКУЛЬТЕТ «Информатика и системы управления»
КАФЕДРА «Информационная безопасность»

ЛАБОРАТОРНАЯ РАБОТА 2:

«Теория игр и исследование операций»

Аналитический и численный методы решения непрерывной выпукло-вогнутой антагонистической игры

ВАРИАНТ: 4

Студент: Гончарова М.К.
(И.О. Фамилия)

Преподаватель: Коннова Н.С.
(И.О. Фамилия)

2020 г.

Цель работы:

Найти оптимальные стратегии непрерывной выпукло-вогнутой антагонистической игры аналитическим и численным методами.

Постановка задачи:

Задана игра со следующими параметрами:

a	b	c	d	e
-15	$\frac{20}{3}$	40	-12	-24

Функция ядра имеет вид:

$$H(x,y) = ax^2 + by^2 + cxy - dx - ey.$$

Найдите цену игры и оптимальные чистые стратегии обоих игроков аналитическим и численным методами. Сравните полученные результаты.

Ход работы:

Аналитическое решение:

Проверка выполнимости условий для принадлежности игры к классу выпукло-вогнутых:

$$H_{xx} = 2a = -30 < 0$$

$$H_{yy} = 2b = \frac{40}{3} > 0$$

Так как $H_{xx} < 0$ и $H_{yy} > 0$ то представленная игра является выпукло − вогнутой.

Найдём производные функции ядра по каждой переменной:

$$H_x = 2ax + cy + d = -30x + 40y - 12,$$

$$H_y = 2by + cx + e = \frac{40}{3}y + 40x - 24.$$

После приравнивания производных к нулю получим:

$$x = -\frac{cy + d}{2a} = -\frac{40y - 12}{-30} = \frac{20y - 6}{15},$$

$$y = -\frac{cx + e}{2b} = -\frac{40x - 24}{\frac{40}{3}} = -\frac{15x - 9}{5}.$$

Учитывая, что x и y должны быть неотрицательными, для оптимальных стратегий соответственно имеем:

$$\psi(y) = \begin{cases} \frac{20y - 6}{15}, y \geq \frac{3}{10}, \\ 0, \quad y < \frac{3}{10}; \end{cases}$$

$$\varphi(x) = \begin{cases} -\frac{15x - 9}{5}, x \leq \frac{3}{5}, \\ 0, \quad x > \frac{3}{5}. \end{cases}$$

Найдём общее решение этих уравнений. Для этого подставим в выражение для y выражение для x:

$$y = -\frac{15x - 9}{5} = -\frac{15\left(\frac{20y - 6}{15}\right) - 9}{5} = -\frac{20y - 15}{5} = -4y + 3,$$

$$5y = 3,$$

$$y = \frac{3}{5}.$$

Подставим получившееся значение y в выражение для x:

$$x = \frac{20y - 6}{15} = \frac{12 - 6}{15} = \frac{2}{5}.$$

Получившееся значение x подставим в выражение для y, чтобы выполнить проверку:

$$y = -\frac{15x - 9}{5} = -\frac{6 - 9}{5} = \frac{3}{5}.$$

Найдём седловую точку игры, подставив получившиеся значения x и y в функцию ядра:

$$H(x,y) = -15 * \left(\frac{2}{5}\right)^2 + \frac{20}{3} * \left(\frac{3}{5}\right)^2 + 40 * \frac{2}{5} * \frac{3}{5} - 12 * \frac{2}{5} - 24 * \frac{3}{5}$$
$$= -\frac{12}{5} + \frac{12}{5} + \frac{48}{5} - \frac{24}{5} - \frac{72}{5} = -\frac{48}{5} = -9{,}6.$$

Далее найдем оптимальные чистые стратегии и цену игры численным методом. Воспользуемся методом аппроксимации функции выигрышей на сетке. Параметр разбиения $N = 1, \dots, 10$. С помощью программы найдем решения при различном шаге сетки для исходной задачи. Ни рисунках 1-5 представлены результаты вычислений.

N=2

(index)	0	1	2
0	0	-10.333	-17.333
1	-9.75	-10.083	-7.083
2	-27	-17.333	-4.333

Есть седловая точка:
x=0.5 y=0.5 H=-10.083

N=3

(index)	0	1	2	3
0	0	-7.259	-13.037	-17.333
1	-5.667	-8.481	-9.815	-9.667
2	-14.667	-13.037	-9.926	-5.333
3	-27	-20.926	-13.37	-4.333

Есть седловая точка:
x=0.333 y=0.667 H=-9.815

Рисунок 1 – первые 2 итерации численного метода

N=4

(index)	0	1	2	3	4
0	0	-5.583	-10.333	-14.25	-17.333
1	-3.938	-7.021	-9.271	-10.688	-11.271
2	-9.75	-10.333	-10.083	-9	-7.083
3	-17.438	-15.521	-12.771	-9.188	-4.771
4	-27	-22.583	-17.333	-11.25	-4.333

Нет седловой точки. Решение методом Брауна-Робинсон: x=0.5 y=0 H=-9.75

N=5

(index)	0	1	2	3	4	5
0	0	-4.533	-8.533	-12	-14.933	-17.333
1	-3	-5.933	-8.333	-10.2	-11.533	-12.333
2	-7.2	-8.533	-9.333	-9.6	-9.333	-8.533
3	-12.6	-12.333	-11.533	-10.2	-8.333	-5.933
4	-19.2	-17.333	-14.933	-12	-8.533	-4.533
5	-27	-23.533	-19.533	-15	-9.933	-4.333

Есть седловая точка:
x=0.4 y=0.6 H=-9.6

Рисунок 2 – вторые 2 итерации численного метода

N=6

(index)	0	1	2	3	4	5	6
0	0	-3.815	-7.259	-10.333	-13.037	-15.37	-17.333
1	-2.417	-5.12	-7.454	-9.417	-11.009	-12.231	-13.083
2	-5.667	-7.259	-8.481	-9.333	-9.815	-9.926	-9.667
3	-9.75	-10.231	-10.343	-10.083	-9.454	-8.454	-7.083
4	-14.667	-14.037	-13.037	-11.667	-9.926	-7.815	-5.333
5	-20.417	-18.676	-16.565	-14.083	-11.231	-8.009	-4.417
6	-27	-24.148	-20.926	-17.333	-13.37	-9.037	-4.333

Нет седловой точки. Решение методом Брауна-Робинсон: x=0.333 y=1 H=-9.667

Рисунок 3 – 5я итерация численного метода

N=7

(index)	0	1	2	3	4	5	6	7
0	0	-3.293	-6.313	-9.061	-11.537	-13.741	-15.673	-17.333
1	-2.02	-4.497	-6.701	-8.633	-10.293	-11.68	-12.796	-13.639
2	-4.653	-6.313	-7.701	-8.816	-9.66	-10.231	-10.531	-10.558
3	-7.898	-8.741	-9.313	-9.612	-9.639	-9.395	-8.878	-8.088
4	-11.755	-11.782	-11.537	-11.02	-10.231	-9.17	-7.837	-6.231
5	-16.224	-15.435	-14.374	-13.041	-11.435	-9.558	-7.408	-4.986
6	-21.306	-19.701	-17.823	-15.673	-13.252	-10.558	-7.592	-4.354
7	-27	-24.578	-21.884	-18.918	-15.68	-12.17	-8.388	-4.333

Есть седловая точка:
x=0.429 y=0.571 H=-9.639

Рисунок 4 – 6я итерация численного метода

N=8

(index)	0	1	2	3	4	5	6	7	8
0	0	-2.896	-5.583	-8.063	-10.333	-12.396	-14.25	-15.896	-17.333
1	-1.734	-4.005	-6.068	-7.922	-9.568	-11.005	-12.234	-13.255	-14.068
2	-3.938	-5.583	-7.021	-8.25	-9.271	-10.083	-10.688	-11.083	-11.271
3	-6.609	-7.63	-8.443	-9.047	-9.443	-9.63	-9.609	-9.38	-8.943
4	-9.75	-10.146	-10.333	-10.313	-10.083	-9.646	-9	-8.146	-7.083
5	-13.359	-13.13	-12.693	-12.047	-11.193	-10.13	-8.859	-7.38	-5.693
6	-17.438	-16.583	-15.521	-14.25	-12.771	-11.083	-9.188	-7.083	-4.771
7	-21.984	-20.505	-18.818	-16.922	-14.818	-12.505	-9.984	-7.255	-4.318
8	-27	-24.896	-22.583	-20.063	-17.333	-14.396	-11.25	-7.896	-4.333

Есть седловая точка:
x=0.375 y=0.625 H=-9.63

Рисунок 5 – 7я итерация численного метода

N=9

(index)	0	1	2	3	4	5	6	7	8	9
0	0	-2.584	-5.004	-7.259	-9.35	-11.276	-13.037	-14.634	-16.066	-17.333
1	-1.519	-3.609	-5.535	-7.296	-8.893	-10.325	-11.593	-12.695	-13.634	-14.407
2	-3.407	-5.004	-6.436	-7.704	-8.807	-9.745	-10.519	-11.128	-11.572	-11.852
3	-5.667	-6.77	-7.708	-8.481	-9.091	-9.535	-9.815	-9.93	-9.881	-9.667
4	-8.296	-8.905	-9.35	-9.63	-9.745	-9.695	-9.481	-9.103	-8.56	-7.852
5	-11.296	-11.412	-11.362	-11.148	-10.77	-10.226	-9.519	-8.646	-7.609	-6.407
6	-14.667	-14.288	-13.745	-13.037	-12.165	-11.128	-9.926	-8.56	-7.029	-5.333
7	-18.407	-17.535	-16.498	-15.296	-13.93	-12.399	-10.704	-8.844	-6.819	-4.63
8	-22.519	-21.152	-19.621	-17.926	-16.066	-14.041	-11.852	-9.498	-6.979	-4.296
9	-27	-25.14	-23.115	-20.926	-18.572	-16.053	-13.37	-10.523	-7.51	-4.333

Нет седловой точки. Решение методом Брауна-Робинсон: x=0.444 y=0.333 H=-9.63

Рисунок 6 – 8я итерация численного метода

N=10

(index)	0	1	2	3	4	5	6	7	8	9	10
0	0	-2.333	-4.533	-6.6	-8.533	-10.333	-12	-13.533	-14.933	-16.2	-17.333
1	-1.35	-3.283	-5.083	-6.75	-8.283	-9.683	-10.95	-12.083	-13.083	-13.95	-14.683
2	-3	-4.533	-5.933	-7.2	-8.333	-9.333	-10.2	-10.933	-11.533	-12	-12.333
3	-4.95	-6.083	-7.083	-7.95	-8.683	-9.283	-9.75	-10.083	-10.283	-10.35	-10.283
4	-7.2	-7.933	-8.533	-9	-9.333	-9.533	-9.6	-9.533	-9.333	-9	-8.533
5	-9.75	-10.083	-10.283	-10.35	-10.283	-10.083	-9.75	-9.283	-8.683	-7.95	-7.083
6	-12.6	-12.533	-12.333	-12	-11.533	-10.933	-10.2	-9.333	-8.333	-7.2	-5.933
7	-15.75	-15.283	-14.683	-13.95	-13.083	-12.083	-10.95	-9.683	-8.283	-6.75	-5.083
8	-19.2	-18.333	-17.333	-16.2	-14.933	-13.533	-12	-10.333	-8.533	-6.6	-4.533
9	-22.95	-21.683	-20.283	-18.75	-17.083	-15.283	-13.35	-11.283	-9.083	-6.75	-4.283
10	-27	-25.333	-23.533	-21.6	-19.533	-17.333	-15	-12.533	-9.933	-7.2	-4.333

Есть седловая точка:
x=0.4 y=0.6 H=-9.6

Рисунок 7 – 9я итерация численного метода

Таким образом, в результате численного метода мы получили следующие оптимальные чистые стратегии и цену игры:

$$x = 0.4$$

$$y = 0.6$$

$$H = -9.6$$

Решение аналитическим методом и численным методом совпали. В программе было задано округление до третьего значения после запятой. Погрешность составляет $\partial = 0$ %.

Вывод:

В результате выполнения лабораторной работы были найдены оптимальные чистые стратегии и цена игры для непрерывной выпукло-вогнутой антагонистической игры аналитическим и численным методами. Результаты двух методов совпали, при условии, что в программе все значения округлялись до третьего знака после запятой.

ПРИЛОЖЕНИЕ А

```
let braunRobinson = require('../lab1/lab1');

function findSaddlePoint(matrix) {

    let matrixMin = [[]];
    let matrixMax = [[]];
    for (let i = 0; i < matrix.length; i++) {
        matrixMin[i] = [];
        matrixMax[i] = [];
    }

    for (let i = 0; i < matrix.length; i++) {

        for (let j = 0; j < matrix.length; j++) {
            matrixMin[i][j] = undefined;
            matrixMax[i][j] = undefined;
        }
    }

    let idxMin = 0;
    let idxMax = 0;

    for (let i = 0; i < matrix.length; i++) {
        let minEl = matrix[i][0];
        let maxEl = matrix[0][i];

        for (let j = 0; j < matrix.length; j++) {


            if (matrix[i][j] < minEl) {
                minEl = matrix[i][j];
                idxMin = j;

            }
            if (matrix[j][i] > maxEl) {
                maxEl = matrix[j][i];
                idxMax = j;
            }

        }
        matrixMin[i][idxMin] = minEl;
        matrixMax[idxMax][i] = maxEl;
    }
    for (let i = 0; i < matrix.length; i++) {

        for (let j = 0; j < matrix.length; j++) {
            if (matrixMax[i][j] && matrixMin[i][j]) {
                return {
                    value: matrixMax[i][j],
                    x: Number((i / (matrix.length - 1)).toFixed(3)),
                    y: Number((j / (matrix.length - 1)).toFixed(3))

                }
            }


        }
    }
    return 0;

}
```

```
function kernel(x, y) {
    let a = -15;
    let b = 20 / 3;
    let c = 40;
    let d = -12;
    let e = -24;
    return Number((a * x * x + b * y * y + c * x * y + d * x + e * y).toFixed(3));

}

function buildMatrix(N) {
    let matrix = [];

    for (let i = 0; i < N + 1; i++) {
        matrix[i] = [];
    }


    for (let i = 0; i < N + 1; i++) {
        for (let j = 0; j < N + 1; j++) {
            matrix[i][j] = kernel(i / N, j / N);
        }
    }
    console.table(matrix);
    return matrix;

}

function findH(matrix, x, y) {
    let H = 0;
    for (let i = 0; i < x.length; i++) {
        for (let j = 0; j < y.length; j++) {
            H += matrix[i][j] * x[i] * y[j];
        }
    }
    return H;

}

function algorithm() {
    for (let i = 2; i < 11; i++) {
        console.log( '\n'+'N=' + i);
        let matrix = buildMatrix(i);
        let saddlePoint = findSaddlePoint(matrix);
        if (saddlePoint) {
            console.log('Есть седловая точка: ' + '\n' + `x=${saddlePoint.x} ` +
`y=${saddlePoint.y} ` + `H=${saddlePoint.value}`);
        } else {
            let obj = braunRobinson(matrix);
            let H = findH(matrix, obj.x, obj.y);

            let matrixDist = [];
            for (let i = 0; i < matrix.length; i++) {
                matrixDist[i] = [];

            }


            for (let i = 0; i < matrix.length; i++) {
                for (let j = 0; j < matrix.length; j++) {
                    matrixDist[i][j] = Math.abs(matrix[i][j] - H);
                }
```

```
            }

            let minEl = Math.min(...[].concat(...matrixDist));
            let braunRobDesicion = {};
            for (let i = 0; i < matrixDist.length; i++) {
                for (let j = 0; j < matrixDist.length; j++) {
                    if (matrixDist[i][j] === minEl) {
                        braunRobDesicion["x"] = Number((i / (matrix.length -
1)).toFixed(3));
                        braunRobDesicion["y"] = Number((j / (matrix.length -
1)).toFixed(3));
                        braunRobDesicion['value'] = matrix[i][j];

                    }
                }
            }


            console.log('Нет седловой точки. Решение методом Брауна-Робинсон: ' + 'x=' +
braunRobDesicion['x'] + ' y=' + braunRobDesicion['y'] + ' H=' +
braunRobDesicion["value"]);
        }

    }

}

algorithm();
```