МАИ

Лабораторная работа №11

 «Символьная математика. Ряды Фурье»Вариант №14

Факультет робототехнических и интеллектуальных систем

Кафедра «Системы приводов летательных аппаратов»

Выполнил:

Студент группы М7О-114БВ-24

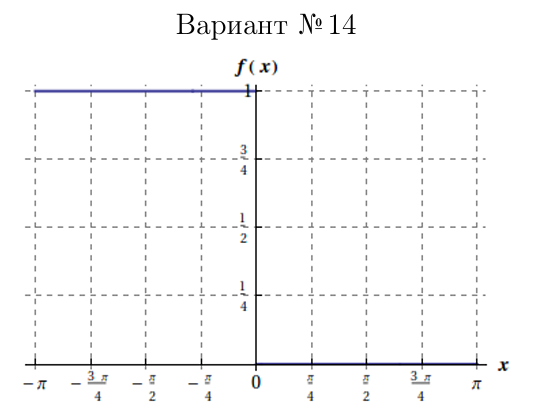
Фельдман Лев Борисович

Проверил:  
Доцент Кафедры 702 Козлова Н.М.

Ассистент Кафедры 702 Милославский Я.Г.

Москва 2025

**Цель работы**: Знакомство с пакетом расширения Symbolic и его использование для автоматизации разложения в ряд Фурье периодической кусочно-линейной функции.



Для периодического продолжения функции

с периодом , её разложение в ряд Фурье на отрезке имеет вид

где коэффициенты вычисляются так:

Из этого видно, что

Следовательно, ряд Фурье принимает компактный вид

При и этот ряд сходится к середине скачка, то есть к ​ (эффект Гиббса). Внутри интервалов и разложение сходится к исходному значению и соответственно.

syms x k;

% Определить кусочную функцию

f\_x = piecewise(x > 0, 0, x < 0, 1);

% Аналитически вычислить коэффициенты Фурье

a0 = (1/pi) \* int(f\_x, x, -pi, pi);

a\_k = (1/pi) \* int(f\_x \* cos(k\*x), x, -pi, pi);

b\_k = (1/pi) \* int(f\_x \* sin(k\*x), x, -pi, pi);

% Выбер набора значений для частичных сумм

N\_values = [1, 2, 4, 8, 16, 32];

colors = lines(numel(N\_values));

% Figure 1: Частичные суммы

figure('Name', 'Частичные суммы рядов Фурье');

hold on;

for idx = 1:numel(N\_values)

N = N\_values(idx);

% Вычислить частичную сумму символически

S\_N = a0/2 + symsum(a\_k\*cos(k\*x) + b\_k\*sin(k\*x), k, 1, N);

% Преобразование в анонимную функцию для построения графика

f\_sum = matlabFunction(S\_N, 'Vars', x);

% Plot частичной суммы

fs = fplot(f\_sum, [-2\*pi, 2\*pi], 'LineWidth', 1.5);

fs.Color = colors(idx, :);

legend\_entries{idx} = sprintf('N = %d', N);

end

% Постройение графика исходной функции для сравнения

xd = [-2\*pi, -pi, -pi, 0, 0, pi, pi, 2\*pi];

yd = [0, 0, 1, 1, 0, 0, 1, 1];

plot(xd, yd, 'k--', 'LineWidth', 2);

legend\_entries{end+1} = 'f(x)';

legend(legend\_entries, 'Location', 'best');

xlabel('x');

ylabel('f\_N(x)');

title('Частичные суммы ряда Фурье для ступенчатой функции');

grid on;

hold off;

% Figure 2: Среднеквадратичное отклонение в зависимости от количества членов

sigma = zeros(size(N\_values));

for idx = 1:numel(N\_values)

N = N\_values(idx);

S\_N = a0/2 + symsum(a\_k\*cos(k\*x) + b\_k\*sin(k\*x), k, 1, N);

err\_sq = (f\_x - S\_N)^2;

sigma(idx) = double(sqrt(int(err\_sq, x, -pi, pi)));

end

figure('Name', 'Среднеквадратичное отклонение');

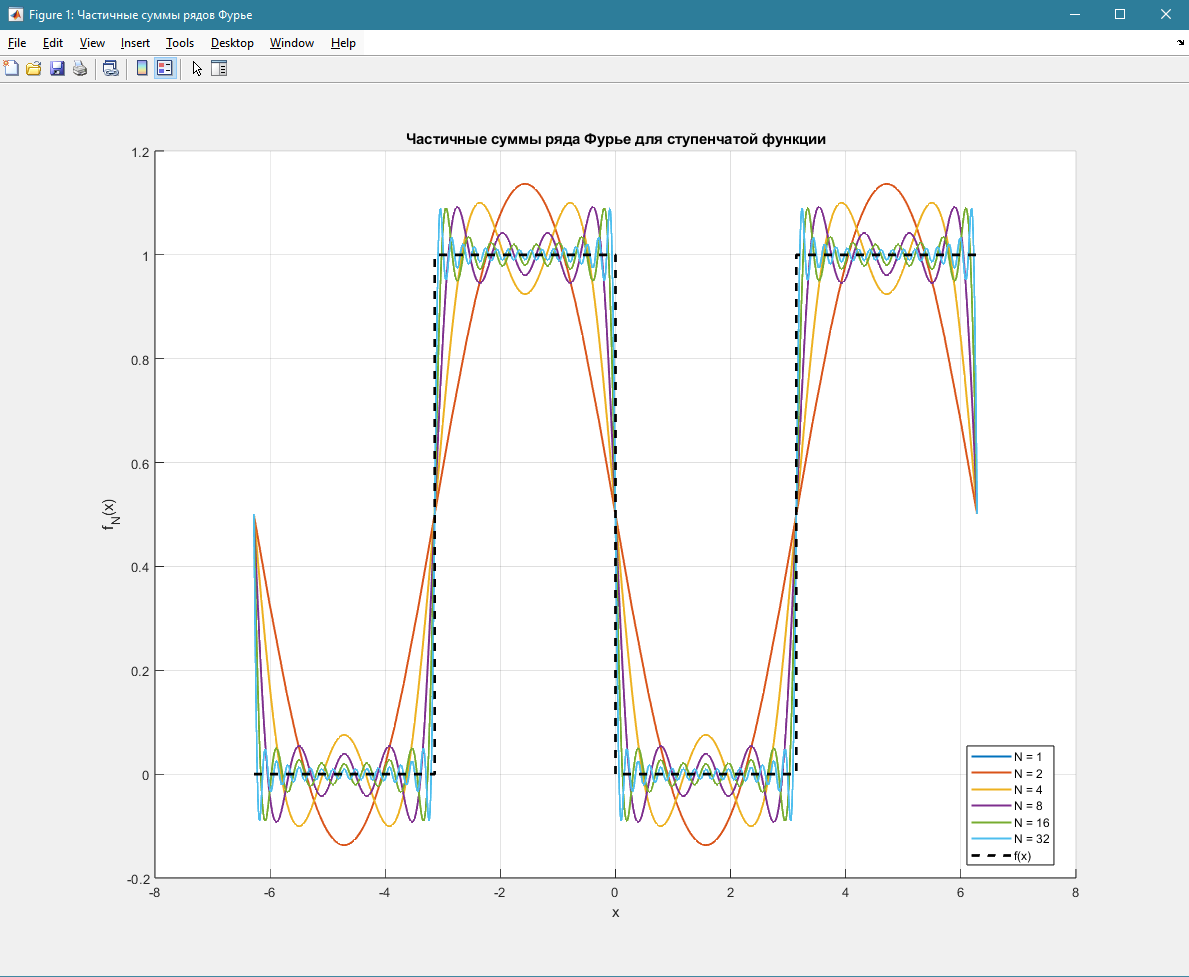
plot(N\_values, sigma, '-o', 'LineWidth', 1.5);

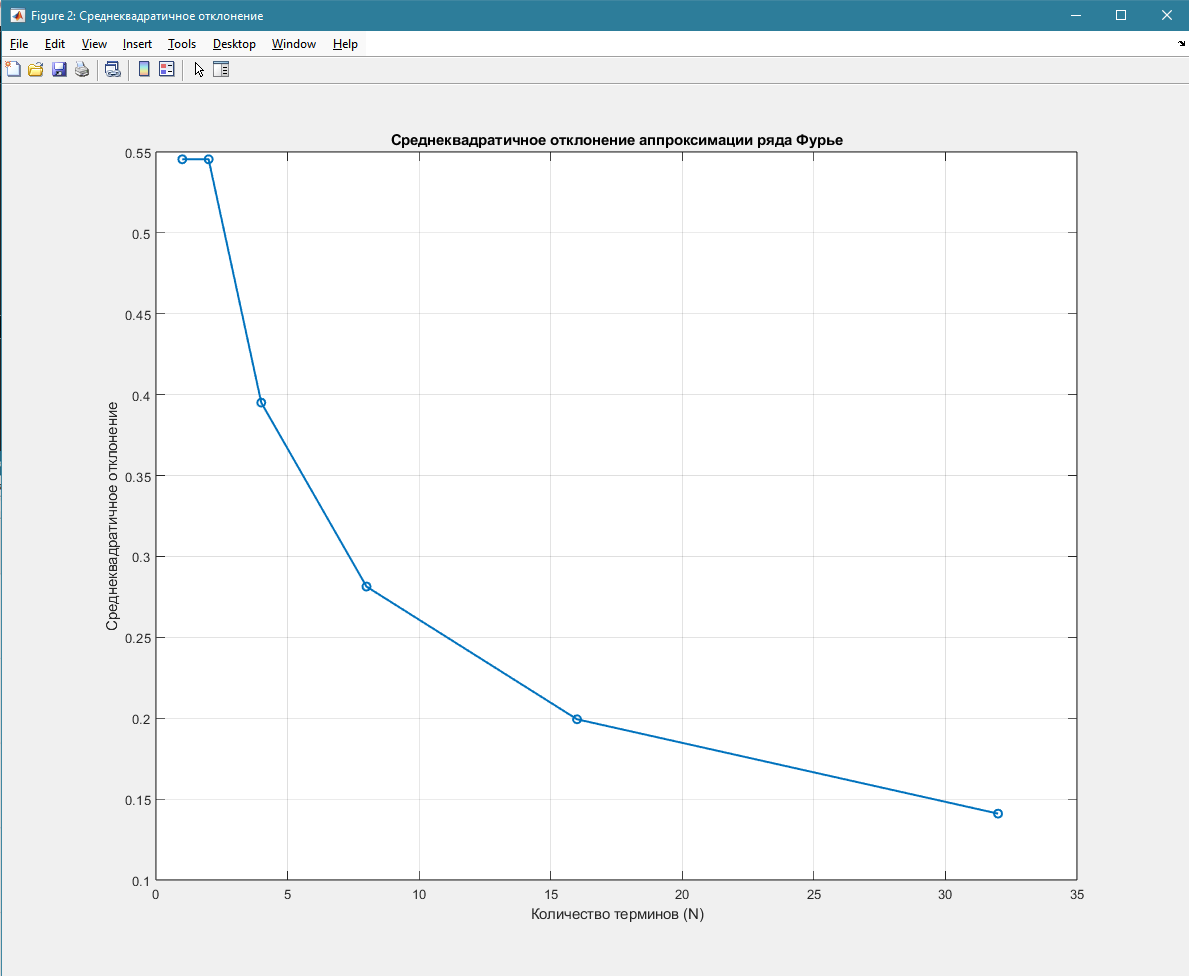
xlabel('Количество терминов (N)');

ylabel('Среднеквадратичное отклонение');

title('Среднеквадратичное отклонение аппроксимации ряда Фурье');

grid on;





## Вывод

В данной работе мы познакомились с пакетом расширения Symbolic и его использованием для автоматизации разложения в ряд Фурье периодической кусочно-линейной функции.

Задание:

Исследовать функцию и построить ее график

% 1. Очистка экрана и задание формата отображения выражений

clc

% 2. Задание символьных переменных

syms x real

syms f real

% 3. Задание функции

fprintf('1. Исследуемая функция.\nf(x)=\n ');

f(x) = (x^2 - x)/(x^2 + x);

pretty(f(x))

% 4. Поиск области определения функции f(x)

fprintf(1,"\n2. Область определения функции f(x).");

N = (x^2 - x);

D = (x^2 + x);

Roots = solve(D==0,x);

fprintf(1,"\nВсё множество вещественных чисел");

if isempty(Roots)

fprintf(1,".\n");

else

fprintf(1," за исключением следующих значений: \n");

disp(Roots');

end

% 5. Определение чётности или нечётности функции

fprintf(1,"\n3. Вид функции.\nФункция f(x) ");

evenFun = @(f, x) isequal(f(x), f(-x));

oddFun = @(f, x) isequal(f(-x), -f(x));

if evenFun(f, x)

fprintf(1, "чётная.\n");

elseif oddFun(f, x)

fprintf(1, "нечётная.\n");

else

fprintf(1, "общего вида.\n");

end

%6. Поиск пересечения точек графика функции с осями Ох и Oy

fprintf(1,"\n4. Точки пересечения.\n");

Roots = solve(f(x),x);

fprintf(1,"C осью Ох ");

if ~isempty(Roots), pretty(Roots);

else fprintf(1,"отсутствуют.\n");

end

fprintf(1,"C осью Оy ");

if any(0, Roots)

F0 = f(0);

if isfinite(F0), pretty(F0);

else fprintf(1,"отсутствует.\n");

end

else fprintf(1,"отсутствует.\n");

end

% 7. Поиск промежутков знакопостоянства

isNumAlwaysPos = isAlways(N > 0);

% Используем solveset для получения множеств решений неравенств

% Решаем для f(x) > 0 и f(x) < 0

PosIntervals = solve(N/D > 0, x,'ReturnConditions', true);

NegIntervals = solve(N/D < 0, x,'ReturnConditions', true);

% Вывод результатов

if ~isempty(PosIntervals)

fprintf('Значение функции f(x) положительно на множестве:\n');

disp(PosIntervals.conditions)

end

if ~isempty(NegIntervals)

fprintf('Значение функции f(x) отрицательно на множестве:\n');

disp(NegIntervals.conditions)

end

%8. Поиск интервалов, на которых функция возрастает и убывает,

% а также экстремумов функции

fprintf("\n6. Первая производная функции f'(x)=\n");

FirstDerivative = simplify(diff(f(x),x));

pretty(FirstDerivative);

fprintf("\n6.1. Промежутки возрастания функции.\n");

UpValInterval = solve(FirstDerivative > 0, x,'ReturnConditions', true);

if UpValInterval == sym("false")

disp("Отсутствуют.");

else

pretty(UpValInterval.conditions);

end

fprintf("\n6.2. Промежутки убывания функции.");

DownValInterval = solve(FirstDerivative < 0, x,'ReturnConditions', true);

if DownValInterval == sym("false")

disp("Отсутствуют.");

else

pretty(DownValInterval.conditions);

end

fprintf("\n6.3. Экстремумы функции.");

Extrema = solve(FirstDerivative == 0,x);

if isempty(Extrema)

fprintf("Отсутствуют.");

else

pretty(Extrema);

end

%9. Поиск промежутков выпуклости и вогнутости,

%нахождение точек перегиба

fprintf("\n7. Вторая производная функции f''(x)=\n");

SecondDerivative = simplify(diff(f(x),x,2));

pretty(SecondDerivative);

fprintf("\n7.1. Промежутки выпуклости функции.\n");

UpCurveInterval = solve(SecondDerivative < 0,x, 'ReturnConditions', true);

if UpCurveInterval == sym("false")

disp("Отсутствуют.");

else

pretty(UpCurveInterval.conditions);

end

fprintf("\n7.2. Промежутки вогнутости функции.\n");

DownCurveInterval = solve(SecondDerivative > 0, x, 'ReturnConditions', true);

if DownCurveInterval == sym("false")

disp("Отсутствуют.");

else

pretty(DownCurveInterval.conditions);

end

fprintf("\n7.3. Точки перегиба функции.\n");

Point = solve(SecondDerivative == 0,x);

if isempty(Point)

disp("Отсутствуют.");

else

pretty(Point);

end

%10. Поиск наклонной асимптоты

fprintf("\n8. Асимптота к графику функции f(x).\n");

% Compute oblique asymptote parameters

K = limit(simplify(f(x)/x), x, inf);

B = limit(simplify(f(x) - K\*x), x, inf);

% Check existence and build symbolic asymptote

if ~isempty(K) && ~isempty(B)

Asymptote = K\*x + B;

fprintf('Asymptote: y = ');

pretty(Asymptote);

else

warning('No oblique asymptote exists.');

Asymptote = [];

end

%11. Построение графика функции f(x)

Hf = ezplot(f(x));

set(Hf,"LineWidth",3,"Color","Red");

XLim = get(gca,"xlim");

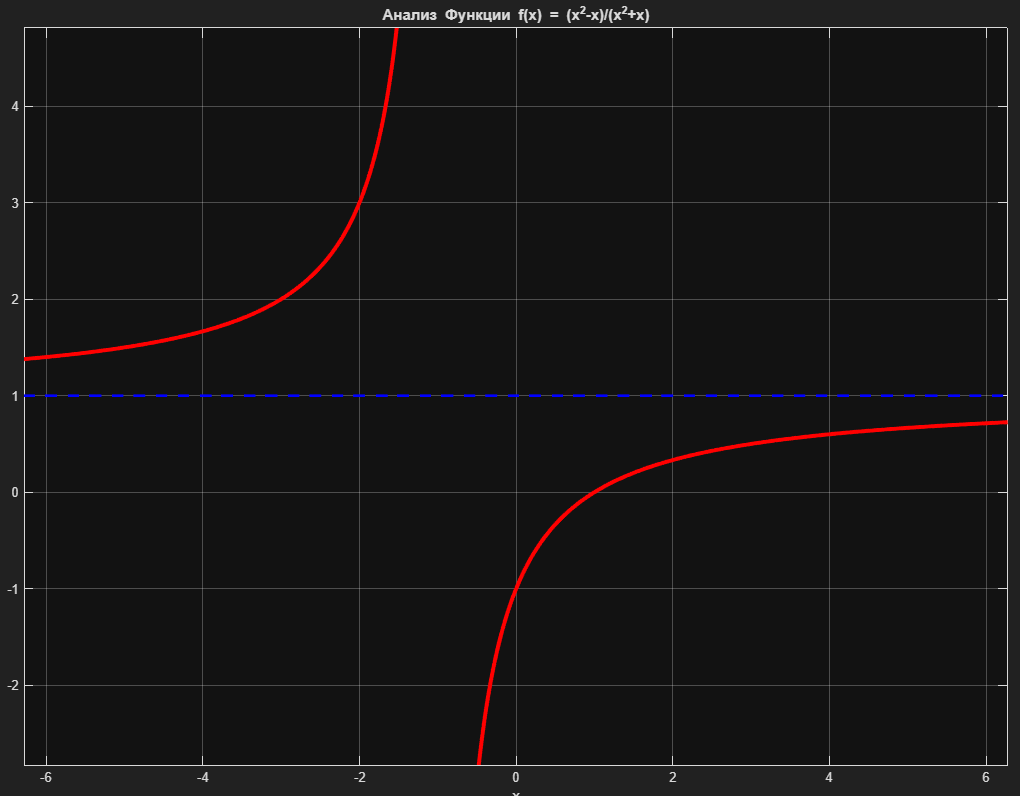
YLim = get(gca,"ylim");

grid on, hold on

HA = ezplot(Asymptote);

set(HA,"LineWidth",2,"Color","Blue","LineStyle","--");

axis([XLim(1) XLim(2) YLim(1) YLim(2)])

  
1. Исследуемая функция.  
f(x)=  
 2  
 x + x  
- --------  
 2  
 x + x  
  
  
2. Область определения функции f(x).  
Всё множество вещественных чисел за исключением следующих значений:   
[-1, 0]  
  
  
3. Вид функции.  
Функция f(x) общего вида.  
  
4. Точки пересечения.

C осью Ох   
1  
C осью Оy отсутствует

Значение функции f(x) положительно на множестве:  
x < -1  
 1 < x  
   
Значение функции f(x) отрицательно на множестве:  
 0 < x & x < 1  
x < 0 & -1 < x  
   
  
6. Первая производная функции f'(x)=  
 2  
--------  
 2  
(x + 1)  
  
  
6.1. Промежутки возрастания функции.

/ x < -1 \  
| |  
\ -1 < x /  
  
6.2. Промежутки убывания функции.

()  
  
6.3. Экстремумы функции.

Отсутствуют.  
7. Вторая производная функции f''(x)=  
 4  
- --------  
 3  
 (x + 1)  
  
7.1. Промежутки выпуклости функции.  
-1 < x  
  
7.2. Промежутки вогнутости функции.

x < -1  
  
7.3. Точки перегиба функции.  
Отсутствуют.  
  
8. Асимптота к графику функции f(x).  
Asymptote: y = 1