12 grudnia 2022 r.

Jakub Grzywaczewski
Grupa nr 2

Porównanie trzech wariantów złożonej kwadratury prostokątów

Projekt nr 47

1 Opis metody

1.1 Kwadratura

Kwadratury są to wzory przybliżonego całkowania, pozwalają nam obliczyć przybliżoną wartość całki $\int_a^b f(x)dx$, czyli pola pod wykresem funkcji f na przedziale (a, b).

Często w fizyce oraz innych naukach ścisłych zajmujących się opisywaniem świata spotykamy się, iż jakaś wielkość fizyczna jest bardzo wygodnie opisana za pomocą całki pewnej funkcji. Jednak te funkcje są bardzo rozbudowane i w większości przypadków nie mają nawet analitycznych rozwiązań. Zatem naszła potrzeba stworzenia metod pozwalających na obliczanie wartości tych oznaczonych całek. Jednym podgatunkiem tych metod (tak zwanego całkowania numerycznego) są właśnie kwadratury.

Opisane wzorem:

$$S(f) = \sum_{k=0}^{n} A_k f(x_k) \tag{1}$$

gdzie

- punkty $x_k \in [a, b]$, dla $k = 0, \ldots, n$, oraz x_k są rosnące nazywane <u>węzłami kwadratury</u>
- A_k, dla $k = 0, \ldots, n$ są <u>współczynnikami kwadratury</u>; Są one niezależne do funkcji f

1.2 Złożona kwadratura prostokątów

Złożona kwadratura prostokątów jest jedną z kwadratur, która wykorzystuje pomysł podzielenia przedziału (a, b) na n przedziałów (najczęściej równoodległych) (definiując tym samym węzły $a = x_0, x_1, \ldots, x_{n-1}, x_n = b$) oraz użycia tych przedziałów do zbudowania prostokątów, których suma będzie przybliżała naszą szukaną całkę.

Złożona kwadratura prostokątów ma 3 główne rodzaje. Rozróżnialne w zależności od punktu przedziału, który zostaje użyty to określenia wysokości prostokątu.

Wzór kwadratury prostokątów dla:
1. Lewego końca przedziału

$$S(f) = \sum_{k=1}^{n} (x_k - x_{k-1}) f(x_{k-1})$$

2. Środka przedziału

$$S(f) = \sum_{k=1}^{n} (x_k - x_{k-1}) f(\frac{x_k + x_{k-1}}{2})$$

3. Prawego końca przedziału

$$S(f) = \sum_{k=1}^{n} (x_k - x_{k-1}) f(x_k)$$

Wzory upraszczają się trochę, kiedy przedziały rzeczywiście są równoodległe. Wtedy przykładowo złożona kwadratura prostokątów na środkach przedziałów można zapisać jako:

$$S(f) = \sum_{k=1}^{n} (x_k - x_{k-1}) f(x_{k-1}) = H * \sum_{k=1}^{n} f(\frac{x_k + x_{k-1}}{2}) \tag{2}$$

gdzie H oznacza długość przedziału zatem $H = \frac{b-a}{n}$

Błąd złożonej kwadratury prostokątów (z węzłem środkowym) jest równy

$$E(f) = \frac{1}{24} H^2 (b - a) f''(\mu)$$

dla pewnego $\mu \in (a, b)$

W owym projekcie sprawdzimy, jak zmieniają się błędy obliczeniowe złożonej kwadratury prostokątów w zależności od wybranego typu owej kwadratury (dla lewego końca / środka / prawego końca przedziałów)

2 Opis funkcjonalności implementacji metody w Matlabie.

Program obliczeniowy składa się z 1 klasy, 2 funkcji, 2 skryptów oraz aplikacji GUI. Teraz opiszę każda z większymi detalami.

2.1 Klasa / Enum - KonceKwad

Ta klasa jest prostą klasą enumeracyjną, która pozwoli nam na wybranie odpowiedniej z 3 kwadratur.

```
classdef KonceKwad
 enumeration
      Lewy, Srodek, Prawy
 end
end
```

2.2 Główna funkcja - kwadProst

Ta funkcja jest sercem całego programu obliczeniowego, ponieważ to właśnie w niej są zaimplementowane wzory kwadratur. Ta funkcja jest odpowiedzialna oszacowywanie pola pod wykresem funkcji i przy okazji zwracam nam także wysokości poszczególnych wysokości jak i z których punktów te wysokości były obliczane.

Sama funkcja ma dokładny opis lecz na potrzeby prezentacji kodu postanowiłem je usunąć.

```
function [wynik, wysokosci, punkty] = kwadProst(wezly, f, koniec, H)
% Sprawdzamy czy zminne sa poprawnej formy
if isscalar(wezly)
    error("Zmienna wezly nie jest wektorem");
end

if ¬isnumeric(wezly)
    error("Wektor welzy nie zawiera wartosci liczbowych")
end

if ¬isa(f, "function_handle")
    error("Zmienna f nie jest referencja do funkcji")
end

if ¬isenum(koniec) || ¬isa(koniec, "KonceKwad")
    error("Nie poprawna zminna koniec. Podaj jedna z opcji klasy ...
        KonceKwad.")
end

wezlyLewe = wezly(1:(end - 1));
wezlyPrawe = wezly(2:end);
```

```
switch koniec
    case KonceKwad.Lewy
        punkty = wezlyLewe;
    case KonceKwad.Prawy
        punkty = wezlyPrawe;
    case KonceKwad.Srodek
        punkty = (wezlyLewe +  wezlyPrawe) / 2;
   otherwise
      error("Zmienna koniec nie jest poprawnie zdefiniowana. U yj ...
         klasy enumeracyjnej KonceKwad.")
end

wysokosci = f(punkty);

if nargin < 4
    elementSumy = (wezlyPrawe - wezlyLewe) * wysokosci;
    wynik = sum(elementSumy);
else
    wynik = H * sum(wysokosci);
end

end
```

Funkcja ta przyjmuje jak widać 4 argumenty:

- wezly - wektor węzłów do użycia w przybliżaniu
- f - funkcja pod całkowa
- koniec - która z 3 kwadratur wybieramy. Jest to instancja klasy KonceKwad.
- H - opcjonalny argument. Kiedy podany przyjmujemy, iż węzły są równo odległe, co przyśpiesza obliczenia

Natomiast zwracana jest trójka:

- wynik - Wynik całkowania numerycznego (skalar)
- wysokości - Wektor poszczególnych prostokątów użytych do obliczenia pola
- punkty - Wektor punktów użytych do obliczenia wysokości

2.3 Funkcja obliczająca błędy - bledyKwad

Również istotna funkcja programu, która dla danego wektora wybranych wartości n (liczby pod-przedziałów na które został podzielony oryginalny przedział (a, b)) oblicza wyniki całkowania numerycznego metodą złożonych kwadratur prostokątów dla 3 rodzajów tej kwadratury.

Tak samo jak w poprzedniej funkcji ukryte są komentarze.

```
function [wyniki, bledy] = bledyKwad(f, F, a, b, n)

if b ≤ a
    error("Przedzial musi miec dodatnia dlugosc");
end

if ¬isnumeric(n)
    error("Wektor n nie zawiera wartosci liczbowych")
end

if ¬isa(f, "function_handle")
    error("Zmienna f nie jest referencja do funkcji")
end

if ¬isa(F, "function_handle")
    error("Zmienna F nie jest referencja do funkcji")
end

dokladny = F(b) - F(a);
H = (b - a) ./ n;

if isscalar(n)

end

wyniki = zeros(length(n), 3);
bledy = zeros(length(n), 3);

for i = 1:length(n)
    wezly = linspace(a, b, n(i));

    [wynikL, ¬, ¬] = kwadProst(wezly, f, KonceKwad.Lewy, H(i));
    [wynikS, ¬, ¬] = kwadProst(wezly, f, KonceKwad.Srodek, H(i));
    [wynikP, ¬, ¬] = kwadProst(wezly, f, KonceKwad.Prawy, H(i));

    wyniki(i, :) = [wynikL, wynikS, wynikP];

    bladL = abs(wynikL - dokladny);
    bladS = abs(wynikS - dokladny);
    bladP = abs(wynikP - dokladny);

    bledy(i, :) = [bladL, bladS, bladP];
end
```

Funkcja ta przyjmuje 5 argumentów:

- f - funkcję pod całkowa
- F - funkcję pierwotną dla funkcji f
- a - początek przedziału
- b - koniec przedziału

- n - liczba pod-przedziałów

Zwraca do użytkownika:

- wyniki - Wektor wyników całkowania numerycznego dla poszczególnych n
- bledy - Wektor błędów bezwzględnych dla poszczególnych n

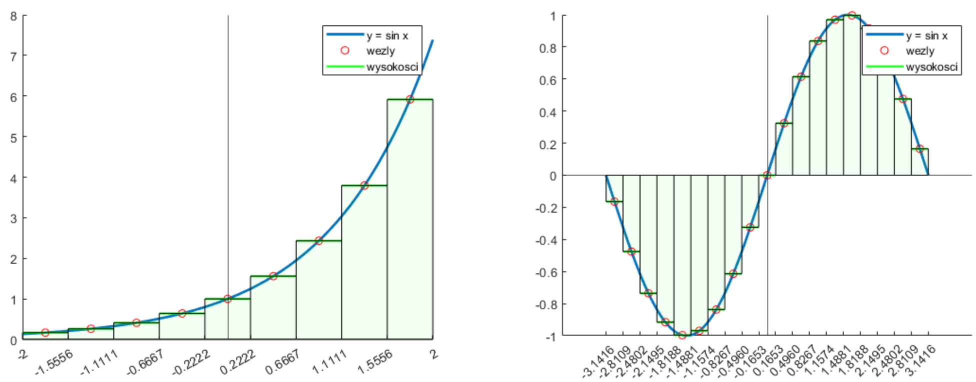
2.4 Prosty skrypt - wizualizacja

Skrypt ten stworzony został tylko do celów wizualizacji w jaki sposób tworzone są prostokąty, których suma pól przybliża nam pole pod wykresem.

```
f = @exp; % Funckja podcalkowa
a = -2; % Pocz tek przedzialu
b = 2; % Koniec przedzialu
n = 10; % Liczba wezlow
koniec = KonceKwad.Srodek; % Ktory koniec

wezly = linspace(a,b,n); % Tworzenie wezlow
H = (b - a) / n; % Wyliczanie odleglosci miedzy wez ami

% Wyznacznie kwadratury
[wynik, wysokosci, punkty] = kwadProst(wezly, f, koniec, H);

% Wartosci dla funkcji podcalkowej
x = linspace(a, b, 1000);
y = f(x);

hold on;
xticks(wezly);

% Plottowanie fukcji podcalkowej
plot(x, y, LineWidth=2);

% Naniesienie wez   w
plot(punkty, wysokosci, "ro")

% Obrazowanie wysokosci i pola kazdego prosotkatow
for i = 2:n

    % Linia wysokosci prostokata
    xx = [wezly(i-1), wezly(i)];
    yy = [wysokosci(i-1), wysokosci(i-1)];
    plot(xx, yy, "g", LineWidth=1.5);

    dlg = wezly(i) - wezly(i - 1);
    wys = abs(wysokosci(i - 1));

    % Pole prostokata
    if wysokosci(i - 1) < 0
        pos = [wezly(i - 1), wysokosci(i - 1), dlg, wys];
    else
```

```
        pos = [wezly(i - 1), 0, dlg, wys];
    end
    rectangle("Position", pos, "FaceColor", [0 1 0 0.05]);

end

% Osie OX i OY
xline(0);
yline(0);

% Utworzenie legendy
legend("y = sin x", "wezly", "wysokosci");
hold off;
```

Przykładowe wykresy wygenerowane przez ten skrypt.



Rysunek 1: Wykresy dla $y = e^x$ oraz $y = \sin x$

2.5 Skrypt obliczeniowy - obliczenia

Skrypt do szybkiego i dokładnego przeglądania wartości i błędów kwadratur prostokątów dla poszczególnych funkcji. Większość z tych funkcji jest bezpośrednio wzięte z aplikacji GUI

```
% "sin x"
% f = @sin;
% F = @(x) -cos(x);
% "sin 2x"
% f = @(x) sin(2 * x);
% F = @(x) (-1/2) * cos(2 * x);
% "cos x"
f = @cos;
F = @(x) sin(x);
% "cos 2x"
% f = @(x) cos(2 * x);
% F = @(x) (1/2) * sin(2 * x);
% "e^x"
% f = @exp;
```

```
% F = @exp;
% "ln x"
% f = @log;
% F = @(x) (x .* log(x)) - x;
% "-ln x"
% f = @(x) -log(x);
% F = @(x) x - (x .* log(x));
% "1/x"
% f = @(x) 1 ./ x;
% F = @(x) log(abs(x));
% "x"
% f = @(x) x;
% F = @(x) (1/2) * (x.^2);
% "x^2"
% f = @(x) x.^2;
% F = @(x) (1/3) * (x.^3);
% "sqrt(x)"
% f = @sqrt;
% F = @(x) (2/3) * (sqrt(x)).^3;
% "1"
% f = @(x) 1;
% F = @(x) x;

a = -2; % Poczatek przedzialu
b = 10; % Koniec przedzialu
n = [5 10 50 100 200 500 1000 10000]; % Ilosci wezlow

[wyniki, bledy] = bledyKwad(f, F, a, b, n);

disp("Wyniki");
disp(wyniki);
disp("Bledy");
disp(bledy);
```

Sama logika skryptu jest bardzo prosta. Większość jego objętości zapewniają wpisane dla wygody pracującego gotowe funkcje, które można analizować pod względem wyznaczania ich pola złożoną kwadraturą prostokątów. Uroszczenie do tej formy owego skryptu, było możliwe dzięki dobrze napisanym wcześniejszym funkcjom.

Przykładowy wynik owego skryptu dla funkcji $y = \cos x$, $a = 0$, $b = \frac{\pi}{2}$ oraz n takiemu jak w kodzie powyżej

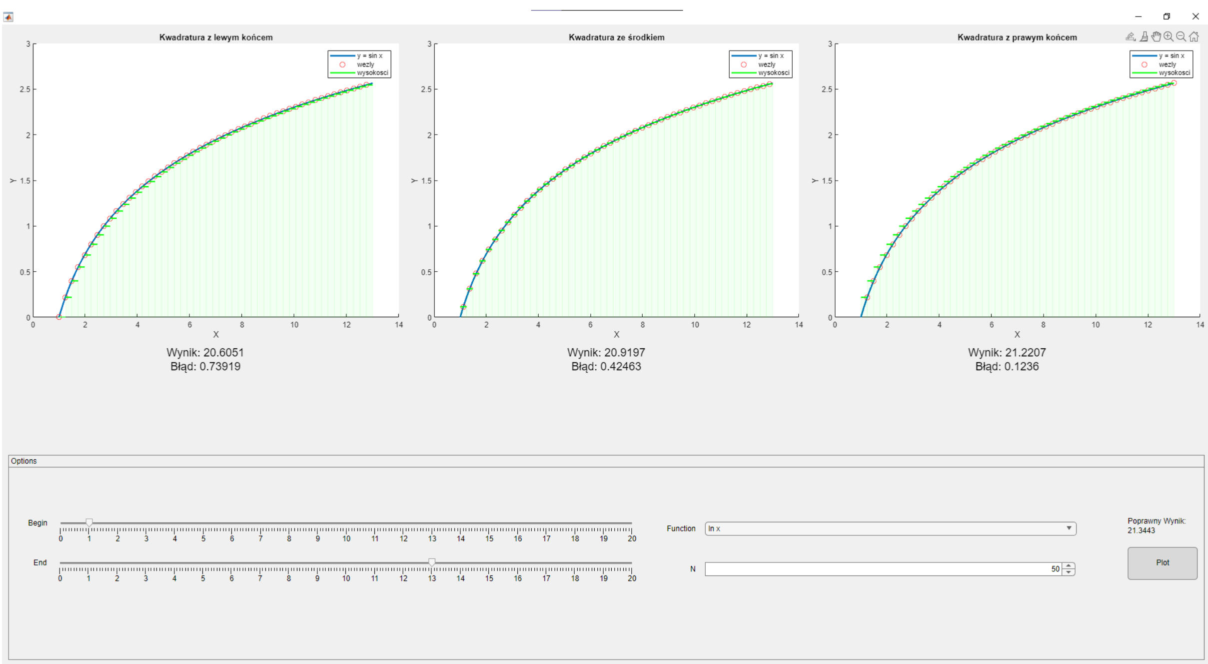
```
    Wyniki
    0.9468    0.8052    0.6326
    0.9763    0.9011    0.8192
    0.9956    0.9800    0.9642
    0.9978    0.9900    0.9821
    0.9989    0.9950    0.9911
    0.9996    0.9980    0.9964
    0.9998    0.9990    0.9982
    1.0000    0.9999    0.9998


```

```
Bledy
     0.0532    0.1948    0.3674
     0.0237    0.0989    0.1808
     0.0044    0.0200    0.0358
     0.0022    0.0100    0.0179
     0.0011    0.0050    0.0089
     0.0004    0.0020    0.0036
     0.0002    0.0010    0.0018
     0.0000    0.0001    0.0002
```

Lewa kolumna to wyniki dla kwadratury z lewym końcem itd.

2.6 Aplikacja - kwadWiz

Jest to aplikacja GUI pozwalająca użytkownikowi szybkie sprawdzenie jak mają się błędy poszczególnych kwadratur do siebie dla wybranych przez niego przedziałów, funkcji oraz licz węzłów.



Rysunek 2: Interfejs aplikacji

W aplikacji jesteśmy w stanie

- Wybrać oba końce przedziałów
- Wybrać funkcje podcałkową z listy dostępnych
- Zaznaczyć ile pod-przedziałów (prostokątów) chcemy użyć
- Wyświetlić wykresy dla każdej z 3 kwadratur
- Odczytać wyniki całkowania numerycznego oraz ich błędy bezwzględne

3 Przykłady obliczeniowe

Przyglądając w podstawowy sposób funkcje (używając aplikacji GUI) i jak dla nich zachowują się poszczególne kwadratury zauważyłem kilka ciekawych przypadków, które potem zadbałem dogłębniej używając skryptu "obliczenia.m".

Podczas obliczeń używałem następujących wartości n: 5 10 50 100 200 500 1000 10000.

3.1 Funkcja nieparzysta na symetrycznym przedziale

Weźmy pod uwagę funkcję, która jest nieparzystą np. $\sin x$, bądź $sin2x$ oraz przedział symetryczny na przykład $(-\pi, \pi)$. Oczekujemy, aby wyniki były jak najbardziej zbliżone do 0.

Dla $\sin x$:

```
Wyniki
   1.0e-15 *

   -0.2790    0.1395    0.1539
   -0.0698   -0.0698    0.0072
    0.0244   -0.0366    0.0401
   -0.0044    0.0453    0.0033
    0.1107    0.0678    0.1146
   -0.1377   -0.0166   -0.1361
   -0.3210   -0.0764   -0.3202
    0.0671    0.0813    0.0671

Bledy
   1.0e-15 *

    0.2790    0.1395    0.1539
    0.0698    0.0698    0.0072
    0.0244    0.0366    0.0401
    0.0044    0.0453    0.0033
    0.1107    0.0678    0.1146
    0.1377    0.0166    0.1361
    0.3210    0.0764    0.3202
    0.0671    0.0813    0.0671
```

Każdy z tych błędów jest straszliwie mały ze względu na naturę sinusa.

Podobnie dla $\sin 2x$:

```
Wyniki
   1.0e-15 *

    0.3078         0   -0.3078
    0.1395         0   -0.1539
   -0.0279   -0.0070   -0.0593
   -0.0035   -0.0453   -0.0187
   -0.0113   -0.0100   -0.0191
```

```
    0.0298    0.1020    0.0267
   -0.0430    0.0211   -0.0446
   -0.0514   -0.0639   -0.0516

Bledy
   1.0e-15 *

    0.3078         0    0.3078
    0.1395         0    0.1539
    0.0279    0.0070    0.0593
    0.0035    0.0453    0.0187
    0.0113    0.0100    0.0191
    0.0298    0.1020    0.0267
    0.0430    0.0211    0.0446
    0.0514    0.0639    0.0516
```

3.2 Funkcja nieparzysta na niesymetrycznym przedziale

Dalej zostańmy przy funkcjach $\sin x$ oraz $\sin 2x$ tylko tym razem użyjmy przedziału $(0, \frac{\pi}{2})$.

Dla $sinx$:

```
Wyniki
    0.6326    0.8052    0.9468
    0.8192    0.9011    0.9763
    0.9642    0.9800    0.9956
    0.9821    0.9900    0.9978
    0.9911    0.9950    0.9989
    0.9964    0.9980    0.9996
    0.9982    0.9990    0.9998
    0.9998    0.9999    1.0000

Bledy
    0.3674    0.1948    0.0532
    0.1808    0.0989    0.0237
    0.0358    0.0200    0.0044
    0.0179    0.0100    0.0022
    0.0089    0.0050    0.0011
    0.0036    0.0020    0.0004
    0.0018    0.0010    0.0002
    0.0002    0.0001    0.0000
```

Oraz dla $\sin 2x$:

```
Wyniki
    0.7584    0.8209    0.7584
    0.8908    0.9046    0.8908
    0.9797    0.9802    0.9797
    0.9899    0.9900    0.9899
    0.9950    0.9950    0.9950
    0.9980    0.9980    0.9980
```

```
     0.9990    0.9990    0.9990
     0.9999    0.9999    0.9999

Bledy
     0.2416    0.1791    0.2416
     0.1092    0.0954    0.1092
     0.0203    0.0198    0.0203
     0.0101    0.0100    0.0101
     0.0050    0.0050    0.0050
     0.0020    0.0020    0.0020
     0.0010    0.0010    0.0010
     0.0001    0.0001    0.0001
```

Wyniki w przypadku $sinx$ różnią się na tyle, iż warto wrzucić owe wartości do tabeli.

Tabela 1: Błędy względne kwadratur przy funkcji $\sin x$

n	Lewy koniec	Środek	Prawy koniec
5	0.3674	0.1948	0.0532
10	0.1808	0.0989	0.0237
50	0.0358	0.0200	0.0044
100	0.0179	0.0100	0.0022
200	0.0089	0.0050	0.0011
500	0.0036	0.0020	0.0004
1000	0.0018	0.0010	0.0002
10000	0.0002	0.0001	0.0000

3.3 Funkcja parzysta na symetrycznym przedziale

Widząc wyniki dla funkcji $\sin x$ możemy postarzeć na jej brata bliźniaka $\cos x$ oraz $\cos 2x$. Zmieńmy natomiast trochę przedział na $(-\frac{\pi}{2}, \frac{\pi}{2})$.

Dla $\cos x$:

```
Wyniki
     1.5169    1.6419    1.5169
     1.7817    1.8092    1.7817
     1.9593    1.9603    1.9593
     1.9798    1.9801    1.9798
     1.9900    1.9900    1.9900
     1.9960    1.9960    1.9960
     1.9980    1.9980    1.9980
     1.9998    1.9998    1.9998

Bledy
     0.4831    0.3581    0.4831
     0.2183    0.1908    0.2183
     0.0407    0.0397    0.0407
```

```
     0.0202    0.0199    0.0202
     0.0100    0.0100    0.0100
     0.0040    0.0040    0.0040
     0.0020    0.0020    0.0020
     0.0002    0.0002    0.0002
```

Dla $\cos 2x$:

```
Wyniki
   1.0e-15 *

     0.1082    0.1395         0
     0.2442    0.2442    0.2093
     0.2721    0.2930    0.2790
     0.0174   -0.0070    0.0419
     0.1413   -0.0436    0.1744
     0.2211    0.4046    0.2072
     0.5047    0.5204    0.5099
     0.2505   -0.3064    0.2510

Bledy
   1.0e-15 *

     0.0142    0.0171    0.1225
     0.1217    0.1217    0.0868
     0.1496    0.1705    0.1566
     0.1050    0.1294    0.0806
     0.0188    0.1661    0.0519
     0.0987    0.2821    0.0847
     0.3822    0.3979    0.3875
     0.1280    0.4289    0.1286
```

3.4 Funkcja parzysta na niesymetrycznym przedziale

Dalej $\cos x$ oraz $\cos 2x$, tylko na przedziale $(0, \frac{\pi}{2}$.

Dla $\cos x$:

```
Wyniki
     0.9468    0.8052    0.6326
     0.9763    0.9011    0.8192
     0.9956    0.9800    0.9642
     0.9978    0.9900    0.9821
     0.9989    0.9950    0.9911
     0.9996    0.9980    0.9964
     0.9998    0.9990    0.9982
     1.0000    0.9999    0.9998

Bledy
     0.0532    0.1948    0.3674
     0.0237    0.0989    0.1808
     0.0044    0.0200    0.0358
```

```
    0.0022    0.0100    0.0179
    0.0011    0.0050    0.0089
    0.0004    0.0020    0.0036
    0.0002    0.0010    0.0018
    0.0000    0.0001    0.0002
```

Tabela 2: Błędy względne kwadratur przy funkcji $\cos x$

n	Lewy koniec	Środek	Prawy koniec
5	0.0532	0.1948	0.3674
10	0.0237	0.0989	0.1808
50	0.0044	0.0200	0.0358
100	0.0022	0.0100	0.0179
200	0.0011	0.0050	0.0089
500	0.0004	0.0020	0.0036
1000	0.0002	0.0010	0.0018
10000	0.0000	0.0001	0.0002

Dla $\cos 2x$:

```
Wyniki
    0.3142    0.0000   -0.3142
    0.1571    0.0000   -0.1571
    0.0314    0.0000   -0.0314
    0.0157    0.0000   -0.0157
    0.0079    0.0000   -0.0079
    0.0031    0.0000   -0.0031
    0.0016    0.0000   -0.0016
    0.0002    0.0000   -0.0002

Bledy
    0.3142    0.0000    0.3142
    0.1571    0.0000    0.1571
    0.0314    0.0000    0.0314
    0.0157    0.0000    0.0157
    0.0079    0.0000    0.0079
    0.0031    0.0000    0.0031
    0.0016    0.0000    0.0016
    0.0002    0.0000    0.0002
```

Warto zauważyć,iż w tym przypadku użycie kwadratury z środkowym węzłem okazuje się najefektywniejsze.

Tabela 3: Błędy względne kwadratur przy funkcji $\cos 2x$

n	Lewy koniec	Środek	Prawy koniec
5	0.3142	0.0000	0.3142
10	0.1571	0.0000	0.1571
50	0.0314	0.0000	0.0314
100	0.0157	0.0000	0.0157
200	0.0079	0.0000	0.0079
500	0.0031	0.0000	0.0031
1000	0.0016	0.0000	0.0016
10000	0.0002	0.0000	0.0002

3.5 Funkcja wykładnicza

Najprościej rozważamy funkcję e^x dla przedziału $(-2, 5)$.

```
Wyniki
   43.6606  104.7364  251.2496
   88.2134  130.1447  192.0078
  135.1799  145.1888  155.9388
  141.6665  146.7645  152.0459
  144.9568  147.5288  150.1465
  146.9458  147.9801  149.0216
  147.6112  148.1292  148.6491
  148.2111  148.2630  148.3149
Bledy
  104.6172   43.5414  102.9718
   60.0645   18.1331   43.7300
   13.0980    3.0890    7.6609
    6.6113    1.5134    3.7681
    3.3210    0.7490    1.8687
    1.3321    0.2978    0.7438
    0.6666    0.1486    0.3713
    0.0667    0.0148    0.0371
```

Tabela 4: Błędy względne kwadratur przy funkcji e^x

n	Lewy koniec	Środek	Prawy koniec
5	104.6172	43.5414	102.9718
10	60.0645	18.1331	43.7300
50	13.0980	3.0890	7.6609
100	6.6113	1.5134	3.7681
200	3.3210	0.7490	1.8687
500	1.3321	0.2978	0.7438
1000	0.6666	0.1486	0.3713
10000	0.0667	0.0148	0.0371

3.6 Funkcja hiperboliczna

Rozważamy funkcję $\frac{1}{x}$ na przedziale $(0.001, 10)$:

```
Wyniki
   1.0e+03 *
    2.0013    0.0027    0.0017
    1.0023    0.0037    0.0025
    0.2043    0.0057    0.0044
    0.1051    0.0064    0.0051
    0.0558    0.0071    0.0058
    0.0267    0.0079    0.0067
    0.0173    0.0084    0.0073
    0.0098    0.0092    0.0088
Bledy
   1.0e+03 *
    1.9921    0.0065    0.0075
    0.9931    0.0055    0.0067
    0.1951    0.0035    0.0048
    0.0959    0.0028    0.0041
    0.0466    0.0021    0.0034
    0.0175    0.0013    0.0025
    0.0081    0.0008    0.0019
    0.0006    0.0000    0.0004
```

4 Analiza wyników

Po przejściu przez funkcje, których współgranie z metodą prostokątów widzimy powyżej oraz więcej, jednym z wniosków jaki jesteśmy w stanie zaobserwować jest fakt, iż błąd dla węzła środkowego jest prawie zawsze najmniejszy, a kiedy nie jest najmniejszy nie odbiega on w dużym stopniu od najlepszego.

Bywały funkcje dla których wybranie przykładowo lewego końca powodowało znaczący wzrost błędu pomiarowego przy stosunkowo małej liczbie węzłów.

Bezpiecznym zatem wyborem na węzeł staje się środek przedziału.

Literatura

[1] Notatki do Metod Numerycznych autorstwa dr. Iwony Wróbel