UNIVERZITA PALACKÉHO V OLOMOUCI
PŘÍRODOVĚDECKÁ FAKULTA

BAKALÁŘSKÁ PRÁCE

Gaussovy kvadraturní formule

Katedra matematické analýzy a aplikací matematiky
Vedoucí bakalářské práce: **Mgr. Jana Burkotová, Ph.D.**
Vypracoval(a): **Martin Veselík**
Studijní program: B1103 Aplikovaná matematika
Studijní obor Matematika–ekonomie se zaměřením na bankovnictví/pojišťovnictví
Forma studia: prezenční
Rok odevzdání: 2018

BIBLIOGRAFICKÁ IDENTIFIKACE

Autor: Martin Veselík

Název práce: Gaussovy kvadraturní formule

Typ práce: Bakalářská práce

Pracoviště: Katedra matematické analýzy a aplikací matematiky

Vedoucí práce: Mgr. Jana Burkotová, Ph.D.

Rok obhajoby práce: 2018

Abstrakt: Gaussovy kvadraturní formule jsou jedním ze způsobů numerického výpočtu integrálu, pokud je příliš složité počítat ho explicitně nebo je to nemožné. Tyto formule vynikají mezi ostatními tím, že dosahují nejvyššího stupně přesnosti. Základem Gaussových kvadratur je znalost ortogonálních polynomů, kterými interpolujeme integrovanou funkci. Podle použitých ortogonálních polynomů se rozlišují jednotlivé typy Gaussových kvadraturních formulí. V této práci jsou představeny Gauss-Legendrovy, Gauss-Čebyševovy, Gauss-Laguerrovy a Gauss-Hermitovy kvadraturní formule. Všechny formule jsou doplněny o kódy v MATLABU.

Klíčová slova: Gaussovy kvadratury, Ortogonální polynomy, Matlab, Lagrangeova interpolace

Počet stran: 60

Počet příloh: 1

Jazyk: český

BIBLIOGRAPHICAL IDENTIFICATION

Author: Martin Veselík

Title: Gaussian quadrature formulas

Type of thesis: Bachelor's

Department: Department of Mathematical Analysis and Application of Mathematics

Supervisor: Mgr. Jana Burkotová, Ph.D.

The year of presentation: 2018

Abstract: Gaussian quadrature formulas are one of the methods of calculating the integral if it is too complex to count it explicitly, or it is impossible. These formulas excel among others by achieving the highest degree of accuracy. The basis of Gaussian quadrature is the knowledge of orthogonal polynomials, interpolating an integrated function. Depending on the orthogonal polynomials used, the individual types of Gaussian quadrature formulas are distinguished. In this work Gauss-Legenders, Gauss-Cebushev, Gauss-Laguer and Gauss-Hermit quadrature formulas are presented. All formulas are matched with MATLAB codes.

Key words: Gaussian quadrature, Orthogonal polynomials, Matlab, Lagrange interpolation

Number of pages: 60

Number of appendices: 1

Language: Czech

Prohlášení

Prohlašuji, že jsem bakalářskou práci zpracoval samostatně pod vedením paní Mgr. Jany Burkotové, Ph.D. a všechny použité zdroje jsem uvedl v seznamu literatury.

V Olomouci dne ....

podpis

Obsah

Poděkování

Rád bych poděkoval paní Mgr. Jany Burkotové, Ph.D. za odborné vedení,za pomoc a rady, trpělivost a ochotu, kterou mi v průběhu zpracování bakalářské práce věnovala.

Úvod

Kvadraturní formule je jednou z metod numerické integrace. V praxi se na numerickou integraci obracíme v případě, že máme integrál

$$I(f) = \int_a^b w(x)f(x)\,\mathrm{d}x,$$

kde $w(x)$je váhová funkce a $f(x)$ integrovaná funkce, pro niž je složité nebo dokonce není možné vypočítat přesnou hodnotu příslušného integrálu.

V této práci se zaměříme na konkrétní typ kvadraturních formulí a to na Gaussovy, které pro daný počet uzlů dosahují nejvyšší stupně přesnosti mezi metodami numerické integrace. Se znalostmi ortogonálních polynomů takovéto fomule lze snadno sestrojit. Nejprve se budeme zabývat pojmem ortogonalita a poté se konkrétně podíváme na 4 typy těchto polynomů, z nichž Legendrovy a Čebyševovy polynomy jsou ortogonální na konečných intervalech a Laguerrovy a Hermitovy polynomy jsou ortogonální na nekonečných intervalech. Každý z těchto polynomů bude doplněn o kód v MATLABU, s jehož pomocí bude možné rychle vypočítat polynom jakéhokoliv řádu a zároveň zjistit i jeho kořeny.

Po seznámení s ortogonálními polynomy si projdeme téma interpolace, tedy proložení hodnot funkce jednodušší funkcí či kombinace funkcí. Konkrétně se zaměříme na polynomiální interpolaci, kdy složitou funkci budeme nahrazovat polynomem. Ukážeme si konstrukci fundamentálních polynomů, jenž budou velmi důležité při určování koeficientů pro kvadraturní formuli. A tvar interpolačního polynomu, kterým budeme nahrazovat integrovanou funkci. Interpolační polynom bude doplněn o kód MATLABU, který ze zadaných uzlů vypočítá interpolační polynom.

V poslední kapitole se zaměříme konkrétně na Gaussovy kvadraturní formule. Řekneme si jakým způsobem vybrat koeficienty a uzly, aby se jednalo o Gaussovy kvadratury a následně se podíváme na určení chyby, v případě, že umíme vypočítat přesnou hodnotu integrálu, budeme řešit skutečnou chybu, pokud nejsme schopni vypočítat přesnou hodnotu integrálu, budeme hledat odhad této chyby. Nakonec si projdeme čtyři typy Gaussových kvadratur, které se od sebe budou lišit použitými ortogonálními polynomy. U každé z formulí si ukážeme obecný tvar, speciální vzhled vzorce pro odhad chyby a speciálně odvozených vzorců pro výpočet koeficientů za pomocí ortogonálních polynomů. Gauss-Legendrovy formule budou doplněny o složený tvar formule a ten následně porovnáme s jinou metodou numerické integrace, konkrétně se složeným Simpsonovým pravidlem. Každý uvedený typ Gaussových kvadratur bude doplněn o příklady, na kterých bude vidět postup výpočtů a určování chyb. Každá formule bude doplněna kódem v MATLABU, který vypočítá podle zadaných vstupů hodnotu kvadraturní formule pro integrovanou funkci.

Kapitola 1

Kvadraturní formule

Kvadraturní formule je jedním ze způsobů, jak přibližně spočítat hodnotu určitého integrálu

$$I(f) = \int_a^b w(x)f(x)\ \mathrm{d}x,$$

kde integrační interval je konečný $a, b \ \in \mathbf{R}$ nebo nekonečný interval $[a, -\infty)$, $(-\infty, \infty)$. Kvadraturní formule se používají tehdy, když máme příliš složitý integrál, který je velmi těžké případně nemožné samostatně vypočítat. Princip aproximace integrálu kvadraturní formulí je odvozen z definice Riemannova integrálu a z jeho geometrického významu. Aproximaci je proto přirozené hledat ve tvaru

$$I(f) \approx \sum_{i=0}^{n} A_i f(x_i),$$

body x_i leží v intervalu $[a, b]$, $i = 0, 1, \ldots, n$ a reálná čísla A_i, $i = 0, 1, \ldots, n$, nezávisí na funkci f.

Tento součet nazveme kvadraturní formulí, jak udává následující definice.

Definice 1.1. *Výraz*

$$Q(f) = \sum_{i=0}^{n} A_i f(x_i) \tag{1.1}$$

nazýváme kvadraturní formulí, čísla A_i, $i = 0, 1, \ldots, n$, koeficienty kvadraturní formule a navzájem různé body x_i, $i = 0, 1, \ldots, n$, uzly kvadraturní formule.

Volba uzlů a koeficientů ovlivňuje přesnost a výpočetní náročnost kvadraturní formule. V případě ekvidistantní sítě uzlů mluvíme o Newton-Cotesových kvadraturních formulích, Lobattova kvadraturní formule je pro případ, kdy máme předepsané některé uzly, pokud je požadavek na stejné koeficienty mluvíme o Čebyševově kvadraturní formuli a pokud je požadavek pouze na nejvyšší přesnost mluvíme o Gaussových kvadraturních formulích. Přesnost kvadraturních formulí se udává pomocí nejvyššího stupně polynomu, který kvadraturní formule aproximuje přesně.

Definice 1.2. *Rozdíl*

$$R(f) = \int_a^b w(x)f(x)\ \mathrm{d}x - \sum_{i=0}^{n} A_i f(x_i) \tag{1.2}$$

budeme nazývat skutečnou chybou kvadraturní formule.

Definice 1.3. *Řekneme, že kvadraturní formule* (1.1) *má stupeň přesnosti N, jestliže*

$$R(x^j) = 0,\ j = 0, 1, \ldots, N,\ R(x^{N+1}) \neq 0$$

Kvadraturní formule mohou být odvozeny z integrace interpolačního polynomu.

Věta 1.1. [1, str. 230] *Kvadraturní formule získaná integrací interpolačního polynomu určeného body $(x_i, f(x_i))$, $i = 0, \ldots, n$, má stupeň přesnosti alespoň n.*

Další věta ukazuje, že správným umístěním uzlů můžeme dosáhnout ještě vyšší přesnosti

Věta 1.2. [1, str. 229] *Kvadraturní formule užívající $n + 1$ uzlů má stupeň přesnosti nejvýše $2n + 1$.*

Nejvyššího stupně přesnosti můžeme dosáhnou použitím kořenů speciálních polynomů, jako uzly kvadraturní formule. V dalším textu budeme symbolem $\prod_j$ označovat množinu všech polynomů stupně j a symbolem $\overline{\prod}_j$ množinu všech normovaných polynomů stupně j, tj. polynomů s koeficientem rovno jedné u nejvyšší mocniny.

Věta 1.3. [1, str. 233] *Nechť* p_n, $p_n \in \overline{\prod}_n$, $n = 0, 1, \ldots$ *tvoří ortogonální systém na intervalu* $[a, b]$ *vzhledem k váhové funkci* w. *Pak tato formule má stupeň přesnosti* $2n + 1$ *právě tehdy, když uzly této kvadraturní formule jsou kořeny polynomu* $p_{n+1} \in \overline{\prod}_{n+1}$

Poznámka 1. *Věta 1.3 je zformulována pro normované polynomy. Ovšem je zřejmé, že volba koeficientů u nejvyšší mocniny neovlivní kořeny polynomu, proto tvrzení zůstává v platnosti i pro nenormované ortogonální polynomy.*

Kapitola 2

Ortogonální polynomy

Pro dosažení maximální přesnosti u kvadraturních formulí je třeba využít jako uzly kořeny ortogonálních polynomů. Pro Gaussovy kvadratury se nejčastěji používají Legendrovy, Čebyševovy, Laguerrovy a Hermitovy polynomy, na které se v této kapitole zaměříme. Každý z polynomů je ortogonální na určitém intervalu s danou váhovou funkcí.

Definice 2.1. *Nechť w je funkce integrovatelná a nezáporná na intervalu $[a, b]$ a $w(x) > 0$ skoro všude na $[a, b]$. Takovou funkci budeme nazývat vahovou funkcí (vahou).*

Samotná ortogonalita je definovaná za pomocí skalárního součinu dvou funkcí, který je tvaru

$$\langle f, g \rangle = \int_a^b w(x) f(x) g(x) \, \mathrm{d}x.$$

Pak o všech funkcích, pro které existuje konečný integrál

$$\int_a^b w(x) f^2(x) \, \mathrm{d}x < +\infty$$

a platí pro ně

$$\langle f, g \rangle = 0,$$

říkáme, že jsou ortogonální na intervalu $[a, b]$ s vahovou funkcí w.

Ortogonální polynomy mají stejné vlastnosti jako jakýkoliv jiný polynom z $\prod_j$, ale narozdíl od ostatních polynomů je lze sestrojit speciálním postupem a to Gram-Schmidtovým ortogonalizačním procesem. Jde o algoritmus, který lze aplikovat na prostor polynomů se skalárním součinem. Vhodně zvolenou váhovou funkcí a mezemi intervalu a, b dostaneme libovolnou sadu ortogonálních polynomů. Například pokud porovnáme tvar Legendrových polynomů 2.1 a Čebyševových polynomů 2.2, můžeme vyvodit, že jejich různou podobu má na svědomí pouze váhová funkce, jelikož jsou vytvořeny na stejných intervalech. V této kapitole se zaměříme na čtyři typy ortogonálních polynomů, z nich dva, Legendrovy 2.1 a Čebyševovy 2.2, jsou ortogonální na konečném intervalu a dva, Laguerrovy 2.3 a Hermitovy 2.4, jsou definovány na nekonečném intervalu.

2.1. Legenderovy polynomy

Legendrovy polynomy lze definovat rekurentním vztahem

$$P_{n+1}(x) = \tfrac{2n+1}{n+1} x P_n(x) - \tfrac{n}{n+1} P_{n-1}(x), \qquad n = 1, 2, 3, \ldots$$

$$P_0 = 1,\ P_1 = x,$$

které jsou ortogonální na intervalu $[-1, 1]$ s vahou

$$w(x) = 1.$$

Pro skalární součin Legendrových polynomů platí [1, str.220]

$$\int_{-1}^{1} P_n(x) P_m(x) \mathrm{d}x = \begin{cases} 0 & \text{pro } n \neq m \\ \frac{2}{2n+1} & \text{pro } n = m \end{cases}$$

V následujícím výpisu je uvedeno prvních pět Legendrových polynomů a v tabulce jsou uvedeny kořeny jednotlivých polynomů.

$$P_0 = 1$$
$$P_1 = x$$
$$P_2 = \frac{3}{2}x^2 - \frac{1}{2}$$
$$P_3 = \frac{5}{2}x^3 - \frac{3}{2}x$$
$$P_4 = \frac{35}{8}x^4 - \frac{15}{4}x^2 + \frac{3}{8}$$
$$P_5 = \frac{63}{8}x^5 - \frac{35}{4}x^3 + \frac{15}{8}x$$

x	P_1	P_2	P_3	P_4	P_5
x_0	0	0,57745	0	-0,8611	0
x_1		-0,5774	-0,7746	0,8611	-0,9062
x_2			0,7746	-0,3400	-0,5385
x_3				0,3400	0,9062
x_4					0,5385

Průběh polynomů je vykreslen na obrázku 2.1.

Pro výpočet kořenů jsem použil MATLAB, konkrétně funkci *roots*, která vypočítá všechny kořeny polynomu, jenž je reprezentován vektorem koeficientů:

např. pro polynom $3x^4 - x^2 + 2x - 9$, bude vektor tvaru [3 0 -1 2 -9].

Pro výpočet polynomů vyšších řádů v MATLABU jsem vytvořil funkci **KLP** - **K**ořeny a **L**egendrovy **P**olynomy.
Vstupem je konstanta **m**, která udává počet kořenů, případně stupěň polynomu.
Výstupem je sloupcový vektor **x** kořenů m-tého polynomu a čtvercová matice **LP**, řádky matice jsou Legendrovy polynomy vyjádřené vektorem koeficientů.
Prvnímu řádku odpovídá polynom P_0 a poslednímu řádku polynom P_m.

```
function [x,LP]=KLP(m)
%Vstup
%m - počet kořenů, které potřebujeme vypočítat,
%neboli stupeň polynomu
%Výstupy
%x - sloupcový vektor kořenů m-tého polynomu
%LP - čtvercová matice o velikosti m+1, řádky matice jsou legendrovy
```

```
%polynomy, vyjádřené vektorem koeficientů
LP=zeros(m+1,m+1);
%tvorba matice do jejíž řádků se budou ukládat koeficienty polynomů
LP(1,m+1)=1; %polynom nultého stupně
LP(2,m)=1; %polynom prvního stupně
i=1;
while i<=m-1 %cyklus pro výpočet polynomů vyšších řádů
r=[(2*i+1)/(i+1) 0];
p1=conv(r,LP(i+1,:));
p1(1)=[];
p0=i/(i+1)*LP(i,:);
LP(i+2,:)= p1 - p0;
i=i+1;
end
x=roots(LP(m+1,:)); %výpočet kořenů posledního polynomu
end
```

Příklad 2.1. *Za pomocí nadefinované funkce **KLP** vypočítáme všechny Legendrovy polynomy až do šestého stupně a kořeny polynomu šestého stupně.*

Funkci zavoláme příkazem:

```
[x,LP]=KLP(6)
```

Výstupem je vektor kořenů polynomu 6-tého stupně

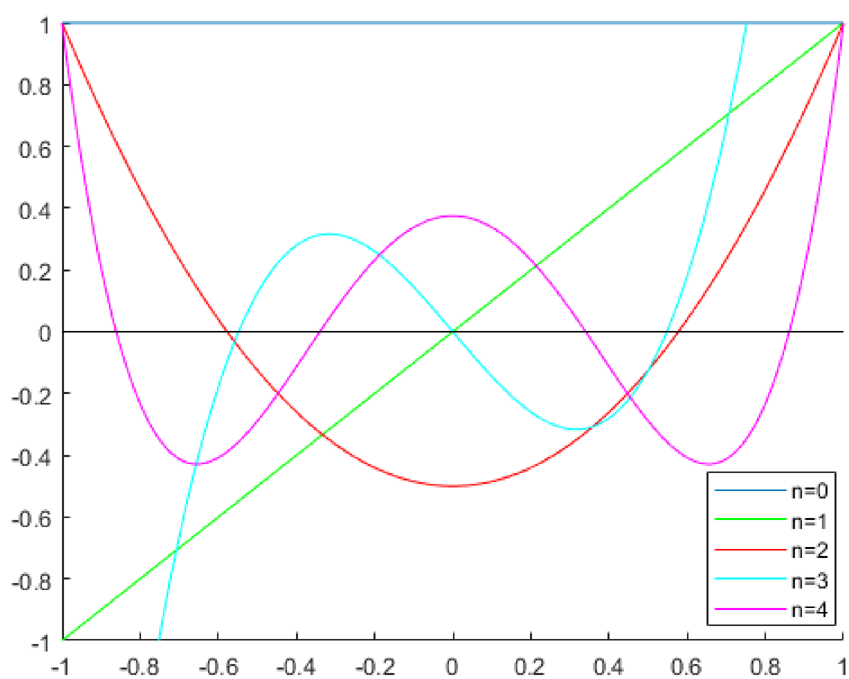
```
x =
   -0.9325
   -0.6612
    0.9325
    0.6612
   -0.2386
    0.2386
```

a matice jejiž řádky odpovídají koeficientům jednotlivých polynomů.

```
LP =

     0          0          0          0          0          0          1
     0          0          0          0          0          1          0
     0          0          0          0          3/2        0         -1/2
     0          0          0          5/2        0         -3/2        0
     0          0          35/8       0         -15/4       0          3/8
     0          63/8       0         -35/4       0          15/8       0
   231/16       0       -315/16       0         105/16      0         -5/16
```

Poslední řádek odpovídá polynomu 6-tého stupně, který je tvaru

$$P_6 = \frac{231}{16}x^6 - \frac{315}{16}x^4 + \frac{105}{16}x^2 - \frac{5}{16}.$$



Obrázek 2.1: Legendrovy polynomy

2.2. Čebyševovy polynomy

Čebyševovy polynomy lze definovat rekurentním vztahem

$$T_{n+1}(x) = 2xT_n(x) - T_{n-1}(x), \qquad n = 1, 2, 3, \dots,$$

$$T_0 = 1,\ T_1 = x,$$

a jsou ortogonální na intervalu [-1,1] s vahou

$$w(x) = \frac{1}{\sqrt{1-x^2}}.$$

Pro skalární součin Čebyševových polynomů platí [1, str.221]

$$\int_{-1}^{1} \frac{T_n(x)T_m(x)}{\sqrt{1-x^2}} \mathrm{d}x = \begin{cases} 0 & \text{pro } n \neq m \\ \pi & \text{pro } n = m = 0 \\ \frac{\pi}{2} & \text{pro } n = m \end{cases}$$

V následujícím výpisu je uvedeno prvních pět Čebyševových polynomů a v tabulce jsou uvedeny kořeny jednotlivých polynomů.

$$\begin{aligned} T_0 &= 1 \\ T_1 &= x \\ T_2 &= 2x^2 - 1 \\ T_3 &= 4x^3 - 3x \\ T_4 &= 8x^4 - 8x^2 + 1 \\ T_5 &= 16x^5 - 20x^3 + 5x \end{aligned}$$

x	T_1	T_2	T_3	T_4	T_5
x_0	0	0,7071	0	-0,9239	0
x_1		-0,7071	-0,8660	0,9239	-0,9511
x_2			0,8660	-0,3827	-0,5878
x_3				0,3827	0,9511
x_4					0,5878

Kořeny v tabulce jsou zaookrouhleny na čtyři desetinná místa, přesné hodnoty je možné dosáhnout klasickým způsobem při výpočtu kořenů polynomu nebo v případě Čebyševových polynomů je možné využít speciálního vzorce [3, str. 315]

$$x_i = cos\left(\frac{2i-1}{2n}\pi\right), \qquad i = 1, \ldots, n.$$

Průběh polynomů je vykreslen na obrázku 2.2.

Pro výpočet polynomů a jejich kořenů v MATLABU jsem vytvořil funkci **KCP** - **K**ořeny a **Č**ebyševovy **P**olynomy.
Vstupem je konstanta **m**, která udává počet kořenů (stupeň polynomu). Výstupem je sloucový vektor **x** kořenů m-tého polynomu a čtvercová matice **CP**, řádky matice jsou Čebyševovy polynomy vyjádřené vektorem koeficientů.

```
function [x,CP]=KCP(m)
%Vstup
%m - počet kořenů které potřebujeme vypočítat,
%neboli stupeň polynomu
%Výstupy
%x - sloupcový vektor kořenů m-tého polynomu
%LP - čtvercová matice o velikosti m+1, řádky matice jsou čebyševovy
%polynomy vyjádřené vektorem koeficientů
CP=zeros(m+1,m+1);
%tvorba matice do jejíž řádků se budou ukládat koeficienty polynomů
%vyjádřené vektorem
CP(1,m+1)=1; %polynom nultého stupně
CP(2,m)=1; %polynom druhého stupně
i=1;
while (i<=m-1) %cyklus pro výpočet polynomů vyšších řádů
r=[2 0];
p0=conv(r,CP(i+1,:));
p0(1)=[];
CP(i+2,:)=p0 - CP(i,:);
i=i+1;
```

```
end
x=roots(CP(m+1,:)); %výpočet kořenů posledního polynomu
end
```

Příklad 2.2. *Za pomocí nadefinované funkce **KCP** vypočítáme všechny Čebyševovy polynomy až do šestého stupně a kořeny polynomu šestého stupně.*

Funkci zavoláme příkazem:

```
[x,CP]=KCP(4)
```

Výstupem je vektor kořenů polynomu 6-tého stupně

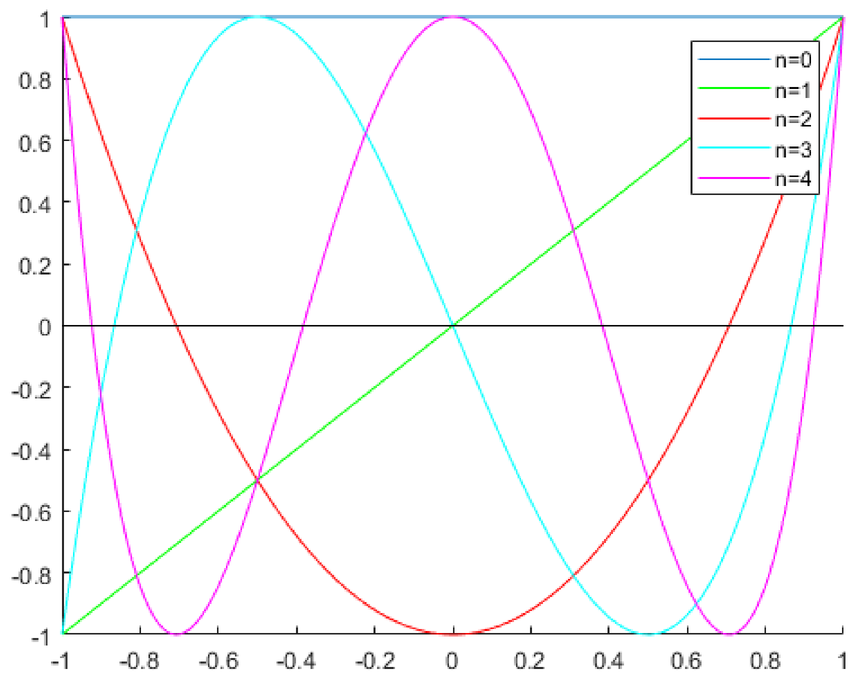
```
x =
   -0.9659
   -0.7071
    0.9659
    0.7071
   -0.2588
    0.2588
```

a matice jejiž řádky odpovídají koeficientům jednotlivých polynomů.

```
CP =
     0     0     0     0     0     0     1
     0     0     0     0     0     1     0
     0     0     0     0     2     0    -1
     0     0     0     4     0    -3     0
     0     0     8     0    -8     0     1
     0    16     0   -20     0     5     0
    32     0   -48     0    18     0    -1
```

Poslední řádek odpovídá polynomu 6-tého stupně, který je tvaru

$$T_6 = 32x^6 - 48x^4 + 18x^2 - 1.$$



Obrázek 2.2: Čebyševovy polynomy

2.3. Laguerrovy polynomy

Laguerrovy polynomy lze definovat rekurentním vztahem

$$L_{n+1}(x,\alpha) = \tfrac{2n+\alpha+1-x}{n+1} L_n(x,\alpha) - \tfrac{n+\alpha}{n+1} L_{n-1}(x,\alpha),\ \ \alpha > -1,\ \ n = 1,2,3,\dots$$

$$L_0 = 1,\ \ L_1 = -x + 1 + \alpha,$$

které jsou ortogonální na intervalu $[0, -\infty)$ s vahou

$$w(x) = x^{\alpha} e^{-x}.$$

Ve výpočtech ortogonality a koeficientů se objevuje funkce gamma $\Gamma(x)$, která je definovaná vztahem

$$\Gamma(x) = \int_0^{\infty} x^{t-1} e^{-t}\ \mathrm{d}x \qquad t > 0.$$

Pro skalární součin Laguerrových polynomů platí [1, str. 221]

$$\int_0^\infty x^\alpha e^{-x} L_n(x,\alpha) L_m(x,\alpha) \mathrm{d}x = \begin{cases} 0 & \text{pro } n \neq m \\ \frac{\Gamma(\alpha+n+1)}{n!} & \text{pro } n = m \end{cases}$$

V následujícím výpisu je uvedeno prvních pět Laguerrových polynomů a v tabulce jsou uvedeny kořeny jednotlivých polynomů pro $\alpha = 0$.

$$L_0 = 1$$
$$L_1 = -x + 1$$
$$L_2 = \tfrac{1}{2}x^2 - 2x + 1$$
$$L_3 = -\tfrac{1}{6}x^3 + \tfrac{3}{2}x^2 - 3x + 1$$
$$L_4 = \tfrac{1}{24}x^4 - \tfrac{2}{3}x^3 + 3x^2 - 4x + 1$$
$$L_5 = -\tfrac{1}{120}x^5 + \tfrac{5}{24}x^4 - \tfrac{5}{3}x^3 + 5x^2 - 5x + 1$$

x	L_1	L_2	L_3	L_4	L_5
x_0	1	3,4142	9,3951	-0,9239	12,6408
x_1		0,5858	4,5366	0,9239	7,0858
x_2			0,4158	1,7458	3,59648
x_3				0,3225	1,4134
x_4					0,2636

Průběh polynomů je vykreslen v obrázku 2.3.

Pro $\alpha = 1$ je uveden stejný počet polynomů a v tabulce jejich kořeny.

$$L_0 = 1$$
$$L_1 = -x + 2$$
$$L_2 = \tfrac{1}{2}x^2 - 3x + 3$$
$$L_3 = -\tfrac{1}{6}x^3 + 2x^2 - 6x + 4$$
$$L_4 = \tfrac{1}{24}x^4 - \tfrac{5}{6}x^3 + 5x^2 - 10x + 5$$
$$L_5 = -\tfrac{1}{120}x^5 + \tfrac{1}{4}x^4 - \tfrac{5}{2}x^3 + 10x^2 - 15x + 6$$

x	L_1	L_2	L_3	L_4	L_5
x_0	2	4,7321	7,7588	10,9539	14,2601
x_1		1,2679	3,3054	5,7312	8,3991
x_2			0,9358	2,57168	4,6108
x_3				0,7433	2,1130
x_4					0,6170

Průběh polynomů na intervalu [0,6] je vykreslen na obrázku 2.4 a na intervalu [6,12] na obrázku 2.5, interval je rozdělený z důvodu přehlednosti, abychom mohli pozorovat kořeny těchto polynomů.

Pro výpočet polynomů a jejich kořenů v MATLABU jsem vytvořil funkci
KLaP - **K**ořeny a **La**guerrovy **P**olynomy.
Vstupem jsou konstanta **m**, která udává počet kořenů, případně stupeň polynomů, a konstanta **a**, což je hodnota α. Výstupem je sloupcový vektor **x** kořenů m-tého polynomu a čtvercová matice **LaP**, řádky matice jsou Laguerrovy polynomy vajádřené vektorem koeficientů.

```
function [x,LaP]=KLaP(m,a)
%Vstupy
%m - počet kořenů, které potřebujeme vypočítat,
%neboli stupeň polynomu
%a - hodnota konstanty alfa
%Výstupy
%x - sloupcový vektor kořenů m-tého polynomu
%LaP - čtvercová matice o velikosti m+1, řádky matice jsou Laguerrovy
%polynomy vyjádřené vektorem koeficientů
LaP=zeros(m+1,m+1);
%tvorba matice do jejíž řádků se budou ukládat koeficienty polynomů
LaP(1,m+1)=1; %polynom nultého stupně
LaP(2,m)=-1;
LaP(2,m+1)=1+a; %polynom prvního stupně
i=1;
```

```
while (i<=m-1) %cyklus pro výpočet polynomů vyšších řádů
p1=[-1 1+2*(i)+a];
p1=conv(p1,LaP(i+1,:));
p1(1)=[];
p0=((i+a))*LaP(i,:);
LaP(i+2,:)= (p1 - p0)/(i+1);
i=i+1;
end
x=roots(LaP(m+1,:)); %výpočet kořenů posledního polynomu
end
```

Příklad 2.3. *Za pomocí nadefinované funkce* ***KLaP*** *vypočítáme všechny Laguerrovy polynomy, až do šestého stupně a kořeny polynomu šestého stupně pro* $\alpha = 1$.

Funkci zavoláme příkazem:

```
[x,LaP]=KLaP5(6,1)
```

Výstupem je vektor kořenů polynomu 6-tého stupně

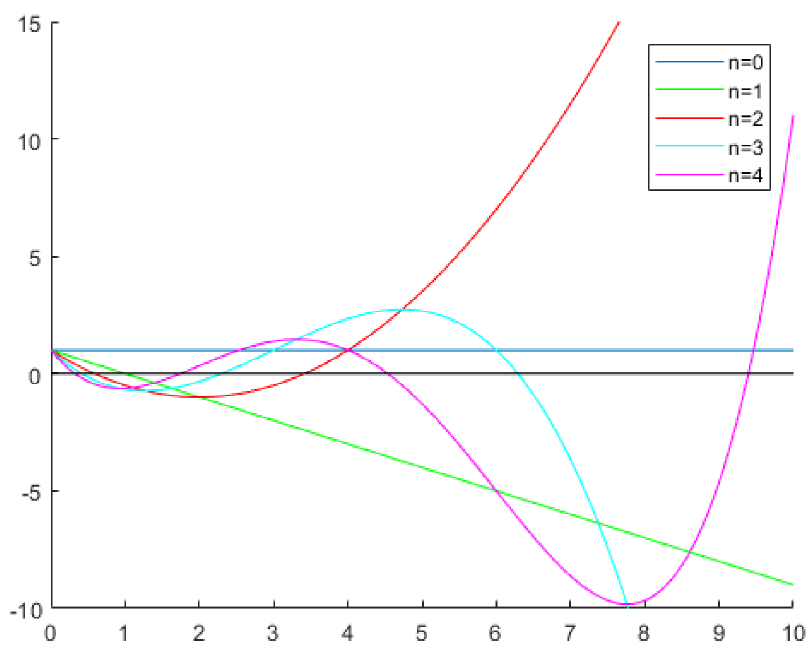
```
x =
   17.6460
   11.2346
    6.9188
    3.8766
    1.7963
    0.5277
```

a matice jejiž řádky odpovídají koeficientům jednotlivých polynomů.

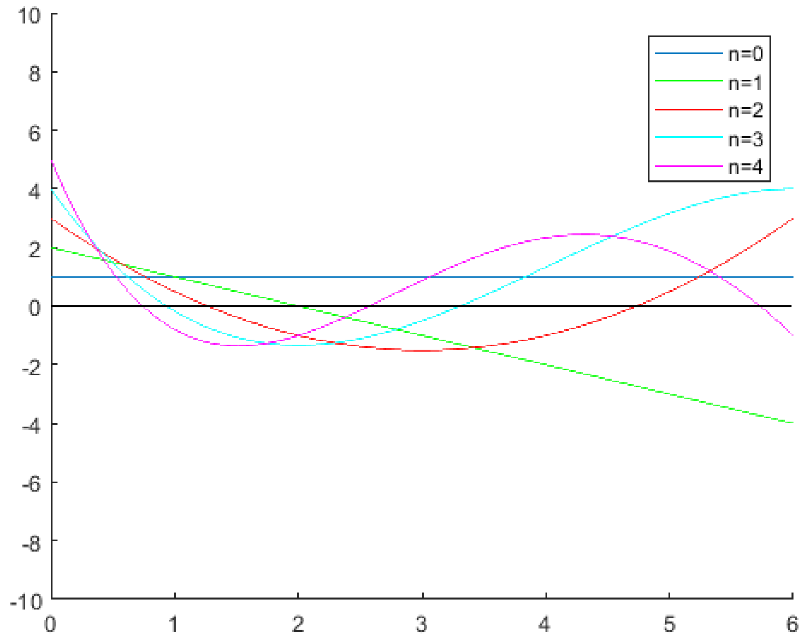
```
LaP =

  0          0          0          0          0          0          1
  0          0          0          0          0         -1          2
  0          0          0          0          1/2       -3          3
  0          0          0         -1/6        2         -6          4
  0          0          1/24      -5/6        5        -10          5
  0         -1/120      1/4       -5/2       10        -15          6
  1/720     -7/120      7/8      -35/6       35/2      -21          7
```

Poslední řádek odpovídá polynomu 6-tého stupně, který je tvaru

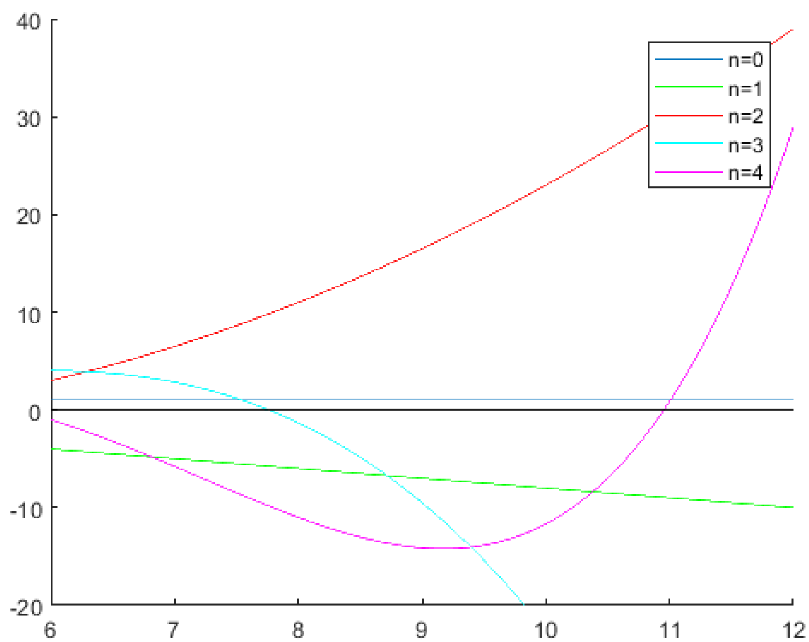
$$L_6 = \frac{1}{720}x^6 - \frac{7}{120}x^5 + \frac{7}{8}x^4 - \frac{35}{6}x^3 + \frac{35}{2}x^2 - 21x + 7.$$



Obrázek 2.3: Laguerrovy polynomy, pro $\alpha = 0$



Obrázek 2.4: Laguerrovy polynomy na intervalu $[0, 6]$, pro $\alpha = 1$



Obrázek 2.5: Laguerrovy polynomy na intervalu $[6, 12]$, pro $\alpha = 1$

Poznámka 2. *V některé literatuře např. [2] se pod názvem Laguerrovy polynomy rozumí ortogonální polynomy na intervalu* $[0, \infty)$ *s vahou* $w(x) = e^{-\alpha x}$*, které se shodují s polynomy definovanými výše pouze pro jednu konkrétní volbu parametru* α*.*

2.4. Hermitovy polynomy

Hermitovy polynomy lze definovat rekurentním vztahem

$$H_{n+1}(x) = 2xH_n(x) - 2nH_{n-1}(x), \qquad n = 1, 2, 3, \dots$$

$$H_0 = 1,\ H_1 = 2x$$

které jsou ortogonální na intervalu $(-\infty, \infty)$ s vahou

$$w(x) = e^{-x^2}.$$

Pro skalární součin Hermitových polynomů platí [1, str.222]

$$\int_{-\infty}^{\infty} e^{-x^2} H_n(x) H_m(x) \mathrm{d}x = \begin{cases} 0 & \text{pro } n \neq m \\ 2^n n! \sqrt{\pi} & \text{pro } n = m \end{cases} . \tag{2.1}$$

V následujícím výpisu je uvedeno prvních pět Hermitových polynomů a v tabulce jsou uvedeny kořeny jednotlivých polynomů.

$$\begin{aligned} H_0 &= 1 \\ H_1 &= 2x \\ H_2 &= 4x^2 - 2 \\ H_3 &= 8x^3 - 12x \\ H_4 &= 16x^4 - 48x^2 + 12 \\ H_5 &= 32x^5 - 160x^3 + 120x \end{aligned}$$

x	H_1	H_2	H_3	H_4	H_5
x_0	0	0,7071	0	-1,6507	0
x_1		-0,7071	1,2247	1,6507	-2,0202
x_2			-1,2247	-0,5246	2,0202
x_3				0,5246	-0,9586
x_4					0,9586

Průběh polynomů je vykreslen na obrázku 2.6.

Pro výpočet polynomů a jejich kořenů v MATLABU jsem vytvořil funkci **KHP** - **K**ořeny a **H**ermitovy **P**olynomy.
Vstupem je konstanta **m**, která udává počet kořenů, případně stupeň polynomu. Výstupem je sloupcový vektor **x** kořenů m-tého polynomu a čtvercová matice **HP**, řádky matice jsou Hermitovy polynomy vyjádřené vektorem koeficientů.

```
function [x,HP]=KHP(m)
%m - počet kořenů, které potřebujeme vypočítat,
%neboli stupeň polynomu
%Výstupy
%x - sloupcový vektor kořenů m-tého polynomu
%HP - čtvercová matice o velikosti m+1, řádky matice jsou Hermitovy
%polynomy vyjádřené vektorem koeficientů
HP=zeros(m+1,m+1);
%tvorba matice do jejíž řádků se budou ukládat koeficienty polynomů
HP(1,m+1)=1; %polynom nultého stupně
HP(2,m)=2; %polynom prvního stupně
i=1;
while i<=m-1 %cyklus pro výpočet polynomů vyšších řádů
r=[2 0];
p1=conv(r,HP(i+1,:));
p1(1)=[];
p0=2*i*HP(i,:);
p = p1 - p0;
HP(i+2,:)=p;
i=i+1;
end
x=roots(HP(m+1,:)); %výpočet kořenů posledního polynomu
end
```

Příklad 2.4. *Za pomocí nadefinované funkce* ***KHP*** *vypočítáme všechny Hermitovy polynomy, až do šestého stupně a kořeny polynomu šestého stupně.*

Funkci zavoláme příkazem:

```
[x,HP]=KHP4(6)
```

Výstupem je vektor kořenů polynomu 6-tého stupně

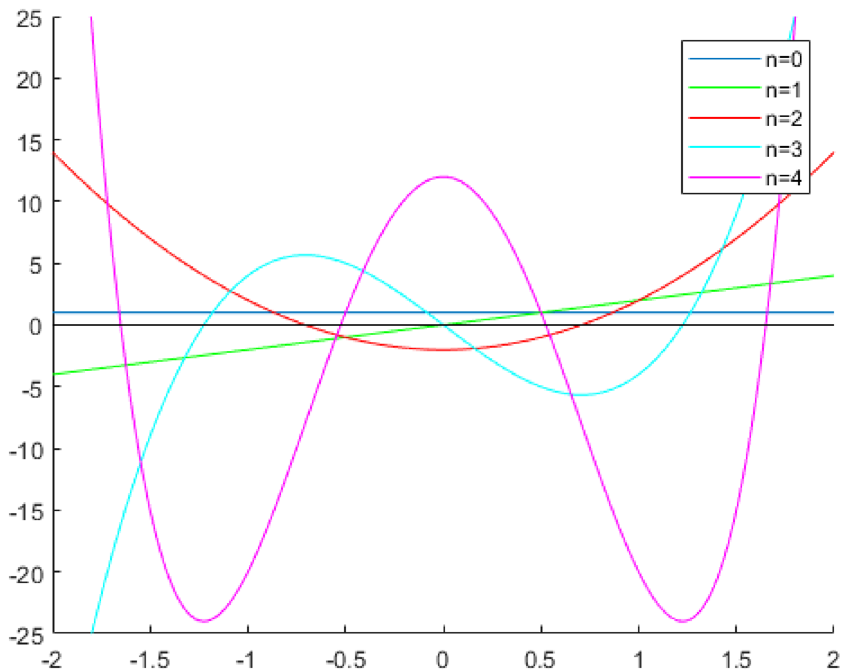
```
x =
   -2.3506
    2.3506
   -1.3358
    1.3358
   -0.4361
    0.4361
```

a matice jejiž řádky odpovídají koeficientům jednotlivých polynomů.

```
HP =
     0     0     0     0     0     0     1
     0     0     0     0     0     2     0
     0     0     0     0     4     0    -2
     0     0     0     8     0   -12     0
     0     0    16     0   -48     0    12
     0    32     0  -160     0   120     0
    64     0  -480     0   720     0  -120
```

Poslední řádek odpovídá polynomu 6-tého stupně, který je tvaru

$$H_6 = 64x^6 - 480x^4 + 720x^2 - 120$$



Obrázek 2.6: Hermitovy polynomy

Kapitola 3

Polynomiální interpolace

V této kapitole se budeme věnovat polynomiální interpolaci, protože základní myšlenkou numerické integrace je nahradit integrand f interpolačním polynomem. Tvar kvadraturní formule, způsob výpočtu koeficientů a vzorec pro odhad chyby je přímo odvozen z integrace interpolačního polynomu.

Úlohou polynomiální interpolace rozumíme:
Na množině $n+1$ navzájem různých bodů (uzlů) x_i najít polynom P_n

$$P_n = a_0 x^n + a_1 x^{n-1} + \cdots + a_n,$$

stupně nejvýše n takový, aby se funkční hodnoty tohoto polynomu a aproximované funkce navzájem rovnaly, tj.

$$f(x_i) = P_n(x_i), \quad i = 0, \ldots, n$$

Účelem interpolace je přibližně určit hodnoty funkce v neznámých bodech a najít odhad mezi přibližnou a přesnou hodnotou.

Konstrukce interpolačního polynomu

Pro zkonsturování interpolačního polynomu je třeba nejdříve sestrojit tzv. fundamentální polynomy l_i, $i = 0, 1, \ldots, n$ s těmito vlastnostmi:

1. l_i je polynom stupně n

2. $l_j = \delta_{jk}$, $j, k = 1, \ldots, n$,

kde δ_{jk} je Kronecovo delta

$$\delta_{jk} = \begin{cases} 0 \text{ pro } j \neq k \\ 1 \text{ pro } j = k \end{cases} .$$

Jelikož $l_j(x)$ musí být polynom, vyplývá z toho, že obsahuje činitele

$$(x - x_0)(x - x_1) \dots (x - x_{j-1})(x - x_{j+1}) \dots (x - x_n)$$

a protože $l_j(a_j) = 1$, můžeme psát

$$l_j(x) = \frac{(x - x_0) \dots (x - x_{j-1})(x - x_{j+1}) \dots (x - x_n)}{(x_j - x_0) \dots (x_j - x_{j-1})(x_j - x_{j+1}) \dots (x_j - x_n)}.$$

Interpolační polynom P_n můžeme pak psát ve tvaru

$$P_n(x) = l_0(x)f(x_0) + l_1(x)f(x_1) + \dots + l_n(x)f(x_n) = \textstyle\sum_{i=0}^{n} l_i(x)f(x_i),$$

interpolační polynom v tomto tvaru se nazývá Lagrangeův interpolační polynom. Jelikož jde o lineární kombinaci polynomů stupně n, je i tento interpolační polynom stupně n. Existence interpolačního polynomu vyplývá z jeho konstrukce, tvrzení následující věty uvádí navíc i jednoznačnost tohoto polynomu.

Věta 3.1. [1, str.159] *Pro $n+1$ daných dvojic čísel*

$$(x_i, f(x_i)) \ i = 0, 1, \dots, n, \ x_i \neq x_k \ \textit{pro} \ i \neq k$$

existuje právě jeden polynom $P_n \in \prod_n$ takový, že

$$P_n(x_i) = f_i, \ \ i = 0, 1, \dots, n. \tag{3.1}$$

Pro výpočet fundamentálních polynomů $l_j(x)$ v MATLABU jsem vytvořil funkci **interpolace**.

Vstupem je sloupcový vektor uzlů interpolace **x** a interpolovaná funkce **f**, jelikož je tento kód naprogramován pro použití u Gaussových kvadratur je vstupem přímo funkce a ne funkční hodnoty. Výstupem je vektor **LIP** koeficientů Lagrangeova interpolačního polynomu

```
function [LIP]=interpolace(x,f)
%Vstup
%x - sloupcový vektor uzlů interpolace
%Výstup
%LFP - vektor koeficientů Lagrangeova interpolačního polynomu
m=length(x);
i=1;
j=1;
jmenovatel=1;
citatel=1;
MAT=zeros(m,m);
%tvorba matice jejíž řádky budou polynomy vyjádřené vektorem
while (i<=m)
while (j<=m)
%cyklusy pro výpočet fundamentálního polynomu pro každý uzel
if (i~=j)
citatel=conv(citatel,[1,-x(j,1)]);
jmenovatel=jmenovatel * (x(i,1)-x(j,1));
end
j=j+1;
end
MAT(i,:)=(f(x(i,1))*citatel)/jmenovatel;
i=i+1;
j=1;
jmenovatel=1;
citatel=1;
end
LIP=sum(MAT);
end
```

Chyba Lagrangeovy interpolace

Jelikož interpolační polynom odpovídá hodnotám funkce f pouze v daných uzlech, zajímá nás, jak moc se interpolační polynom liší od funkce f v ostatních bodech. Proto je třeba se zabývat chybou interpolace $E(x) = f(x) - P_n(x)$.

Věta 3.2. [1, str.167] *Nechť* $f \in C^{(n+1)}$ $[a,b]$ *a nechť uzly* $x_i \in [a,b]$, $i = 0,1,\ldots,n$, $x_i \neq x_k$ *pro* $i \neq k$. *Nechť dále* $P_n \in \prod_n$ *je interpolační polynom splňující podmínky*(3.1). *Pak ke každému bodu* $\overline{x} \in [a,b]$ *existuje bod* $\xi \in (a,b)$ *tak, že platí*

$$E(\overline{x}) = f(\overline{x}) - P_n(\overline{x}) = \frac{\omega_{n+1}(x)}{(n+1)!} f^{(n+1)}(\xi), \qquad \omega_{n+1}(x) = \prod_{i=0}^{n} (\overline{x} - x_i)$$

Odhad chyby hledáme ve tvaru

$$|E(\overline{x})| \leq \frac{M_{(n+1)}}{(n+1)!} |\omega_{n+1}(x)|,$$

kde M je $|f^{(n+1)(x)}| \leq M_{(n+1)}, \forall x \in [a,b]$. Tento odhad závisí na vlastnostech interpolované funkce a na volbě uzlů x_i.

Souvislost mezi Interpolací a Kvadraturní formulí

V případě, že známe Lagrangeův interpolační polynom P_n pro funkci ve známých bodech a víme jak vypadá chyba interpolace E můžeme jednoduše najít souvislost mezi interpolací a kvadraturní formulí. Integrovaná funkce se dá vyjádřit jako součet interpolačního polynomu a chyby interpolace.

$$f(x) = P_n(x) + E(x) = \sum_{i=0}^{n} l_i(x) f(x_i) + \frac{\omega_{n+1}(x)}{(n+1)!} f^{(n+1)}(\xi), \qquad \xi \in [a,b].$$

Pokud rovnici vynásobíme váhovou funkcí $w(x)$, zintegrujeme v mezích a, b a vytkneme konstanty před integrál dostaneme

$$\int_a^b w(x) f(x) \, \mathrm{d}x = \sum_{i=0}^{n} f(x_i) \int_a^b w(x) l_i(x) \, \mathrm{d}x +$$

$$+\frac{1}{(n+1)!}\int_a^b w(x)\omega_{n+1}(x)f^{(n+1)}(\xi)\ \mathrm{d}x = +\sum_{i=0}^{n} f(x_i)A_i + R(f).$$

Kvadraturní formule je tedy tvaru

$$Q(f) = \sum_{i=0}^{n} f(x_i)A_i,$$

kde koeficienty A_i získáme ze vztahu

$$A_i = \int_a^b w(x)l_i(x)\ \mathrm{d}x. \tag{3.2}$$

Pro chybu kvadraturní formule platí

$$R(f) = \frac{1}{(n+1)!}\int_a^b w(x)\omega_{n+1}(x)f^{(n+1)}(\xi)\ \mathrm{d}x.$$

Kapitola 4

Gaussovy kvadratury

Kvadraturní formule, jejichž uzly a koeficienty jsou vybrány tak, aby bylo dosaženo maximálního stupně přesnosti, se nazývají Gaussovy kvadraturní formule. Tato kapitola je zaměřena konkrétně na čtyři rozdílné typy Gaussových kvadratur, které se liší použitými ortogonálními polynomy.
Jak bylo uvedeno výše ve Větě 1.3, Gaussovy kvadratury využívají kořenů ortogonálních polynomů jako uzly, koeficienty jsou vypočítané integrací interpolačního polynomu, díky tomu tyto formule dosahují přesnosti $2n + 1$.

Tvar chyby Gaussovy kvadraturní formule je uveden v následující větě.

Věta 4.1. [1, str. 236] *Nechť $f \in C^{(2n+2)}[a, b]$. Chybu Gaussovy kvadraturní formule lze vyjádřit ve tvaru*

$$R(f) = \tfrac{f^{(2n+2)}(\eta)}{(2n+2)!} \int_a^b w(x)\omega_{n+1}^2(x)\ \mathrm{d}x, \quad \eta \in (a, b),$$

kde $\omega_{n+1}(x) = (x - x_0)\dots(x - x_n)$.

Jelikož není možné určit bod η, je vhodné absolutní hodnotu chybu ohraničit zhora. V tom případě nahradíme čitatel $f^{(2n+2)}(\eta)$ horní mezí funkce M_{2n+2}, pro kterou platí $|f^{(2n+2)}(\eta)| \leq M_{2n+2}$. Vztah pro odhad chyby bude tvaru

$$|R(f)| \leq \frac{M_{2n+2}}{(2n+2)!} \int_a^b w(x)\omega_{n+1}^2(x)\ \mathrm{d}x. \tag{4.1}$$

4.1. Gauss-Legendrovy kvadraturní formule

V případě konstantní váhové funkce na konečném intervalu se pro numerický výpočet používá Gauss-Legendrových kvadraturních formulí. Formule je tvaru

$$\int_{-1}^{1} f(x)\ \mathrm{d}x = \sum_{i=0}^{n} A_i f(x_i) + R(f),$$

kde x_i je i-tý kořen Legendrova ortogonálního polynomu n-tého stupně.
V případě Gauss-Legendrovy formule je možné použít pro výpočet chyby speciální vzorec. [2, str.391]

$$R(f) = \frac{2^{2n+3}((n+1)!)^4}{(2n+3)[(2n+2)!]^3} f^{2n+2}(\xi)$$

Lineární transformace

V případě, že máme jiný konečný interval než $[-1, 1]$ je třeba před použitím Gaussovy kvadratury transformovat tento interval do požadované podoby.
Postup transfomace

$$\int_{a}^{b} f(t)\ \mathrm{d}t = \frac{b-a}{2} \int_{-1}^{1} f\left(\frac{b-a}{2}x + \frac{a+b}{2}\right)\ \mathrm{d}x.$$

Koeficienty této formule je možné spočítat integrací interpolačního polynomu (3.2), tento způsob je vhodný pro ruční výpočet.

Příklad 4.1. *Užitím Gauss-Legendrovy kvadraturní formule pro m=3, aproximujte integrál*

$$\int_{1}^{3} e^x sin\ x\ \mathrm{d}x$$

Jelikož m=3 je třeba tří uzlů. Pro výpočet těchto uzlů musíme nejdřív vypočítat Legendrův polynom třetího stupně, ten je roven

$$P_3 = \frac{5}{2}x^3 - \frac{3}{2}x$$

a následně, spočítáme kořeny polynomu třetího stupně.

$$x_0 = 0,\ x_1 = \sqrt{\frac{3}{5}},\ x_2 = -\sqrt{\frac{3}{5}}$$

Dále je třeba spočítat koeficienty kvadraturní formule,

$$l(x_0) = \frac{\left(x - \sqrt{\frac{3}{5}}\right)\left(x + \sqrt{\frac{3}{5}}\right)}{\left(0 - \sqrt{\frac{3}{5}}\right)\left(0 + \sqrt{\frac{3}{5}}\right)} = \frac{x^2 - \frac{3}{5}}{-\frac{3}{5}} = -\frac{5}{3}x^2 + 1$$

$$A_0 = \int_{-1}^{1} -\frac{5}{3}x^2 + 1 = \left[-\frac{5}{9}x^3 + x\right]_{-1}^{1} = \left(-\frac{5}{9} + 1\right) - \left(\frac{5}{9} - 1\right) = \frac{8}{9}$$

$$l(x_1) = \frac{(x-0)\left(x + \sqrt{\frac{3}{5}}\right)}{\left(\sqrt{\frac{3}{5}} - 0\right)\left(\sqrt{\frac{3}{5}} + \sqrt{\frac{3}{5}}\right)} = \frac{x^2 + \sqrt{\frac{3}{5}}x}{\frac{6}{5}} = \frac{5}{6}x^2 + \sqrt{\frac{5}{12}}x$$

$$A_1 = \int_{-1}^{1} \frac{5}{6}x^2 + \sqrt{\frac{5}{12}}x = \left[\frac{5}{18}x^3 + \sqrt{\frac{5}{48}}x^2\right]_{-1}^{1} = \left(\frac{5}{18} + \sqrt{\frac{5}{48}}\right) - \left(-\frac{5}{18} + \sqrt{\frac{5}{48}}\right) = \frac{5}{9}$$

$$l(x_2) = \frac{(x-0)\left(x - \sqrt{\frac{3}{5}}\right)}{\left(-\sqrt{\frac{3}{5}} - 0\right)\left(-\sqrt{\frac{3}{5}} - \sqrt{\frac{3}{5}}\right)} = \frac{x^2 - \sqrt{\frac{3}{5}}x}{\frac{6}{5}} = \frac{5}{6}x^2 - \sqrt{\frac{5}{12}}x$$

$$A_2 = \int_{-1}^{1} \frac{5}{6}x^2 - \sqrt{\frac{5}{12}}x = \left[\frac{5}{18}x^3 - \sqrt{\frac{5}{48}}x^2\right]_{-1}^{1} = \left(\frac{5}{18} - \sqrt{\frac{5}{48}}\right) - \left(-\frac{5}{18} - \sqrt{\frac{5}{48}}\right) = \frac{5}{9}$$

Jelikož, formule nefunguje na integrál s mezemi $a = 1, b = 3$, musíme použít transformaci.

$$t_0 = \left(\frac{b-a}{2}x_0 + \frac{a+b}{2}\right) = x_0 + 2 = 2$$

$$t_1 = \left(\frac{b-a}{2} x_1 + \frac{a+b}{2} \right) = x_1 + 2 = 2,7746$$

$$t_2 = \left(\frac{b-a}{2} x_2 + \frac{a+b}{2} \right) = x_2 + 2 = 1,2254$$

Následně je ještě třeba celou formuli vynásobit hodnotou $\frac{b-a}{2} = 1$.

Přibližná hodnota integrálu pro $m = 3$ je tedy rovna

$$\int_1^3 e^x sin\ x \ \mathrm{d}x \approx A_0 f(t_0) + A_1 f(t_1) + A_2 f(t_2) = 10,9484.$$

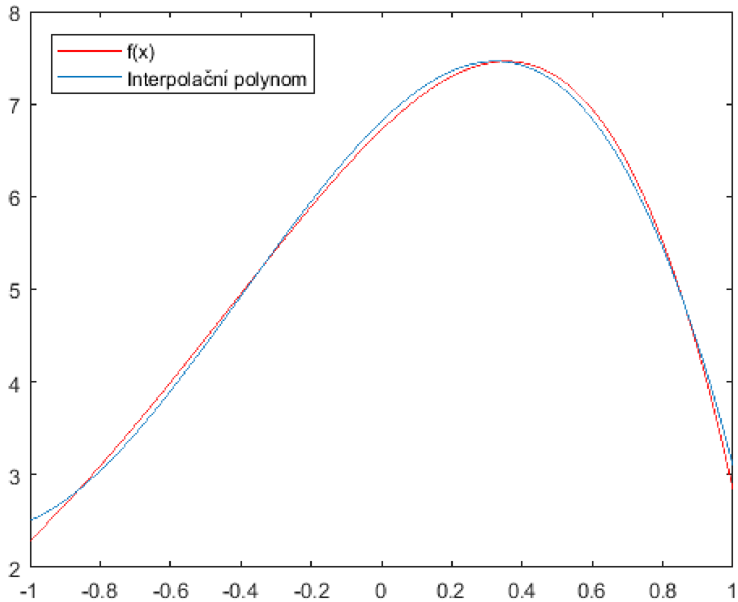
Následně je třeba určit chybu, u tohoto integrálu můžeme vypočítat jak odhad chyby tak i skutečnou chybu. Pro výpočet skutečné chyby je nutné vypočítat integrál dvakrát pomocí metody per partes:

$$\int_1^3 e^x sin\ x = \left[e^x \frac{sin\ x - cos\ x}{2} \right]_0^3 = e^3 \frac{sin\ 3 - cos\ 3}{2} - e \frac{sin\ 1 - cos\ 1}{2} \doteq 10,9502$$

Skutečnou chybu určíme jednoduchým rozdílem podle vztahu (1.2) a odhad chyby provedeme pomocí (4.1).

Skutečná chyba: $R(f) = 1,7677 \cdot 10^{-3}$
Odhad chyby: $R(f) = 1,0100 \cdot 10^{-2}$

Na obrázku 4.1 můžeme vidět průběh funkce $f(x)$ a průběh interpolačního polynomu pro $x_0 = 0,\ x_1 = \sqrt{\frac{3}{5}},\ x_2 = -\sqrt{\frac{3}{5}}$.



Obrázek 4.1: průběh funkce f a interpolačního polynomu z Příkladu 4.1

Další možností výpočtu koeficientů je pomocí speciálně odvozeného vzorce [1, str.237]:

$$A_i = \frac{2(1-x_i)^2}{(n+1)^2(P_n(x_i))^2} \qquad i = 0, 1, \dots, n, \tag{4.2}$$

kde P_n je Legendrův ortogonální polynom n-tého stupně.

Příklad 4.2. *Pomocí vzorce vypočítejte koeficienty pro tříbodovou Gauss-Legendrovu formuli*

$$A_0 = \frac{2(1-x_0)^2}{(m)^2(\frac{5}{2}x_0^3 - \frac{3}{2}x_0)^2} = 0,8889$$

$$A_1 = \frac{2(1-x_1)^2}{(m)^2(\frac{5}{2}x_1^3 - \frac{3}{2}x_1)^2} = 0,5556$$

$$A_2 = \frac{2(1-x_2)^2}{(m)^2(\frac{5}{2}x_2^3 - \frac{3}{2}x_2)^2} = 0,5556$$

Pro výpočet koeficientů v MATLABU jsem vytvořil funkci

KGL - **K**oeficienty **G**auss-**L**egendrovy formule.

Vstupem je konstanta **m**, která udává počet kořenů, případně supeň polynomu.

Výstupem je sloupcový vektor **A** koeficientů kvadraturní formule.

```
function A=KGL(m)
%Vstup
%m - počet koeficientů(případně počet kořenů polynomu)
%Výstup
%A - sloupcový vektor koeficientů kvadraturní formule
[x,LP]=KLP(m);
%naprogramovaná funkce pro výpočet kořenů
%a polynomů potřebných k výpočtu
i=1;
A=zeros(m,1);
while i<=m
%cyklus pro výpočet jednotlivých koeficientů
A(i,1)=2*(1-(x(i,1))^2)/((m^2)*((polyval(LP(m,:),x(i,1)))^2));
i=i+1;
end
```

Pro výpočet hodnoty kvadraturní formule jsem vytvočil funkci

GLKF - **G**auss-**L**egendrova **K**vadraturní **F**ormule.

Vstupem jsou konstanta **m**, která udává počet kořenů, případně stupeň polynomu, funkce **f**, integrovaná funkce, **a**/**b**, dolní/horní mez integrálu.

```
function int=GLKF(m,f,a,b)
%Vstupy
%m - počet kořenů, neboli stupeň polynomu
%f - integrovaná funkce
%a/b - dolní/horní mez integrálu
```

```
%Výstup
%int - hodnota kvadraturní formule
if a<b && a~=-Inf && b~=Inf
[x,~]=KLP(m);
A=KGL(m);
if a~=-1 || b~=1
x=(x*(b-a)/2)+((a+b)/2);
t=(b-a)/2;
else
t=1;
end
x=f(x);
int=sum(A.*x)*t;
else
string("Špatně zadané meze")
end
```

Gauss-Legendrovy formule mají velmi důležitou vlastnost, tak jak uvádí [1, str. 249] a [2, str. 412], s rostoucím počtem uzlů konverguje posloupnost Gauss-Legendrových formulí k přesné hodnotě integrálu.

Tuto vlastnost si můžeme ověřit například u příkladu 4.1, v následující tabulce je hodnota inegrálu vypočteného numericky pomocí nadefinované funkce **GLKF** a chyba vypočítaná pomocí vztahu (1.2) pro $m = 1, \dots, 7$.

m	Hodnota integrálu	Skutečná Chyba
1	13,4377	2,4875
2	11,1415	$1{,}9132 \cdot 10^{-1}$
3	10,9484	$1{,}7677 \cdot 10^{-3}$
4	10,9501	$3{,}0221 \cdot 10^{-5}$
5	10,9502	$8{,}6610 \cdot 10^{-8}$
6	10,9502	$6{,}5156 \cdot 10^{-10}$
7	10,9502	$9{,}1838 \cdot 10^{-13}$

Příklad 4.3. *Užitím Gauss-Legendrovy kvadraturní formule pro m=4, aproximujte integrál*

$$\int_0^4 arctg(\sqrt{x})\ \mathrm{d}x$$

Jelikož m=4, je třeba čtyř uzlů. Pro výpočet těchto uzlů musíme nejdřív odvodit Legendrův polynom čtvrtého stupně, ten je roven

$$P_4 = \frac{35}{8}x^4 - \frac{15}{4}x^2 + \frac{3}{8}$$

a následně, spočítáme kořeny polynomu třetího stupně.

$$x_0 = -0,8611,\ x_1 = 0,8611,\ x_2 = -0,3400,\ x_3 = 0,3400$$

Dále je třeba spočítat koeficienty kvadraturní formule, které spočítáme použitím již odvozeného vzorce

$$A_0 = \frac{2(1-x_0)^2}{(m)^2(\frac{35}{8}x_0^4 - \frac{15}{4}x_0^2 + \frac{3}{8})^2} = 0,3479$$

$$A_1 = \frac{2(1-x_1)^2}{(m)^2(\frac{35}{8}x_1^4 - \frac{15}{4}x_1^2 + \frac{3}{8})^2} = 0,3479$$

$$A_2 = \frac{2(1-x_2)^2}{(m)^2(\frac{35}{8}x_2^4 - \frac{15}{4}x_2^2 + \frac{3}{8})^2} = 0,6521$$

$$A_3 = \frac{2(1-x_3)^2}{(m)^2(\frac{35}{8}x_3^4 - \frac{15}{4}x_3^2 + \frac{3}{8})^2} = 0,6521$$

Jelikož, formule nefunguje na integrál s mezemi $a = 0, b = 4$, musíme použít transformaci.

$$t_0 = \left(\frac{b-a}{2}x_0 + \frac{a+b}{2}\right) = 2x_0 + 2 = 0,2778$$

$$t_1 = \left(\frac{b-a}{2}x_1 + \frac{a+b}{2}\right) = 2x_1 + 2 = 3,7222$$

$$t_2 = \left(\frac{b-a}{2}x_2 + \frac{a+b}{2}\right) = 2x_2 + 2 = 1,3200$$

$$t_3 = \left(\frac{b-a}{2}x_3 + \frac{a+b}{2}\right) = 2x_3 + 2 = 2,6800$$

Dále je ještě třeba celou formuli vynásobit hodnotou $\frac{b-a}{2} = 2$.

Přibližná hodnota integrálu pro $m = 4$ je tedy rovna

$$\int_0^4 arctg(\sqrt{x})\ \mathrm{d}x \approx 2(A_0 f(t_0) + A_1 f(t_1) + A_2 f(t_2) + A_3 f(t_3)) \doteq 3,5584$$

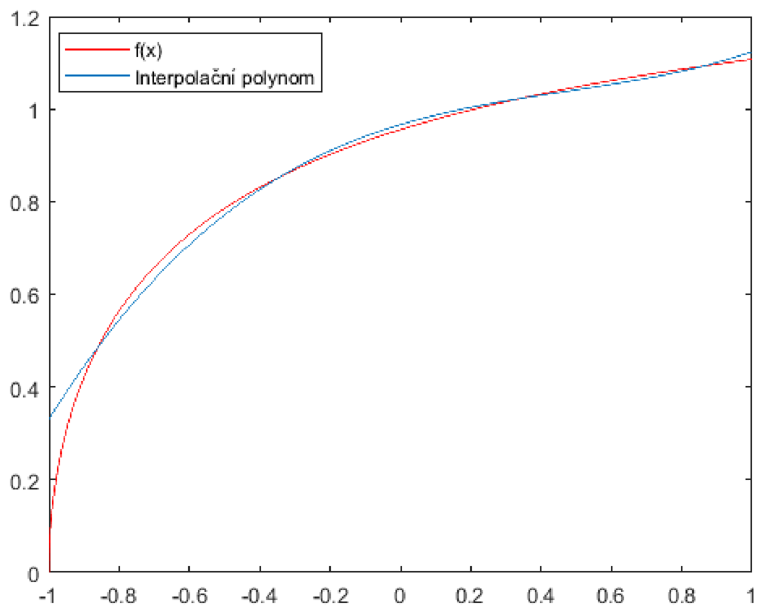
Následně je třeba určit chybu, u tohoto integrálu můžeme vypočítat pouze skutečnou chybu, jelikož derivace této funkce není spojitá nemůžeme vzorec (4.1) použít na odhad chyby. Pro určení skutečné je třeba vypočítat integrál pomocí substituce a metody per partes.

$$\int_0^4 arctg(\sqrt{x})\ \mathrm{d}x = \left[arctg(\sqrt{x})(x+1) - \sqrt{x}\right]_0^4 = 5\ arctg(2) - 2 \doteq 3,5357$$

Skutečnou chybu určíme jednoduchým rozdílem podle vztahu (1.2). V následující tabulce je uvedena hodnota integrálu spočítaného numericky a skutečná chyba pro $m = 1, \ldots, 5$

m	Hodnota integrálu	Skutečná Chyba
1	3,8213	$2{,}8552 \cdot 10^{-1}$
2	3,6029	$6{,}7149 \cdot 10^{-2}$
3	3,5584	$2{,}2704 \cdot 10^{-2}$
4	3,5458	$1{,}0007 \cdot 10^{-2}$
5	3,5410	$5{,}2771 \cdot 10^{-3}$

V obrázku 4.2 můžeme vidět průběh funkce $f(x)$ a průběh interpolačního polynomu pro $x_0 = -0,8611,\ x_1 = 0,8611,\ x_2 = -0,3400,\ x_3 = 0,3400$.



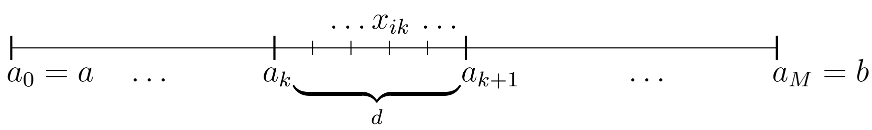
Obrázek 4.2: průběh funkce f a interpolačního polynomu z Příkladu 4.3

4.1.1. Složené Gauss-Legendrovy formule

Myšlenkou složených formulí je rozdělit interval na menší části a na každou z nich použít metodu kvadraturních formulí. Díky tomu získáme při stejném počtu uzlů mnohem přesnější hodnoty.

Původní interval rozdělíme na M podintervalů, které označíme např. $I_k,\ k = 1,\dots,M,\ I_k = (a_{k-1}, a_k)$, kde a_{k-1} a a_k jsou krajní body jednotlivých podintervalů, $d = \frac{b-a}{M}$ je délka podintervalu a platí $a_0 = a,\ a_k = a + k\frac{b-a}{M},\ k = 0,\dots,M,\ a_M = b$.

Na obrázku 4.3 můžeme vidět rozdělení integračního intervalu na M stejně velkých podintervalů.



Obrázek 4.3: Rozdělení integračního intervalu

Hodnota integrálu se pak rovná součtu jednotlivých integrálů přes podintervaly I_k.

$$\int_a^b f(x)\ \mathrm{d}x = \int_{a_0}^{a_1} f(x)\ \mathrm{d}x + \int_{a_1}^{a_2} f(x)\ \mathrm{d}x + \dots + \int_{a_{M-1}}^{a_M} f(x)\ \mathrm{d}x$$

Na každý z těchto integrálů budeme aplikovat Gauss-Legendrovu formuli, přičemž použijeme transformaci 4.1.

Složená Gauss-Legendrova formule bude pak mít tvar

$$\mathbf{Q} = \sum_{k=0}^{M-1} \sum_{i=0}^{n} A_i f(x_{ik})$$

kde x_{ik} je i-tý uzel v k-tém intervalu.

Pro výpočet hodnoty kvadraturní formule jsem vytvořil funkci
SGLKF - **S**ložená **G**auss-**L**egendrova **K**vadraturní **F**ormule.
Vstupem jsou konstanta **m**, která udává počet uzlů(kořenů) pro jednotlivé části intervalu, funkce **f**, integrovaná funkce, **a**/**b**, dolní/horní mez celého integrálu a konstanta **M**, která udává na kolik podintervalů se má interval rozdělit.

```
function int=SGLKF(m,f,a,b,M)
%Vstupy
%m - počet uzlů (kořenů) pro jednotlivé části intervalu
%f - integrovaná funkce
%a/b - dolní a horní mez integrálu
%M - na kolik dílů se má interval rozdělit
```

```
%Výstup
%int - hodnota celkové kvadraturní formule
int=0;
d=(b-a)/M; %určení kroku
c=a+d;
j=1;
A=KGL(m);
while j<=M
%cyklus pro jednotlivé části intervalu
[x,~]=KLP(m);
if a~=-1 || c~=1
%lineární transformace
x=(x*(c-a)/2)+((a+c)/2);
t=(c-a)/2;
else
t=1;
end
g=sum(f(x).*A);
int=int+g*t;
j=j+1;
a=a+d;
c=c+d;
end
end
```

Příklad 4.4. *Pro příklad 4.1 použijeme složenou Gauss-Legendrovu kvadraturní formuli. Interval rozdělíme na 4 části a na každém podintervalu vypočítáme integrál pomocí tříbodové Gaussovy kvadratury.*

Integrál bude tedy tvaru

$$\int_1^3 e^x sin\ x\ \mathrm{d}x = \int_1^{\frac{3}{2}} e^x sin\ x\ \mathrm{d}x + \int_{\frac{3}{2}}^2 e^x sin\ x\ \mathrm{d}x + \int_2^{\frac{5}{2}} e^x sin\ x\ \mathrm{d}x + \int_{\frac{5}{2}}^3 e^x sin\ x\ \mathrm{d}x$$

Jednotlivé kvadraturní formule pro každý integrál budeme označovat $Q_1,\ Q_2,\ Q_3,\ Q_4$. Jelikož pro každý z integrálů použijeme tři uzly a tři koeficienty, budou pro všechny stejné.

$$x_0 = 0,\ x_1 = 0,7746,\ x_2 = -0,7746$$

$$A_0 = \frac{8}{9},\ A_1 = \frac{5}{9},\ A_2 = \frac{5}{9}$$

Teď je třeba pro jednotlivé části použít lineární transformaci a vypočítat hodnotu kvadraturní formule

$$t_0 = \frac{\frac{3}{2}-1}{2}x_0 + \frac{\frac{3}{2}+1}{2} = 1,25000$$

$$t_1 = \frac{\frac{3}{2}-1}{2}x_1 + \frac{\frac{3}{2}+1}{2} = 1,44365$$

$$t_2 = \frac{\frac{3}{2}-1}{2}x_2 + \frac{\frac{3}{2}+1}{2} = 1,05635$$

$$Q_1 = \frac{\frac{3}{2}-1}{2}(A_0 e^{t_0} sin\ t_0 + A_1 e^{t_1} sin\ t_1 + A_2 e^{t_2} sin\ t_2) \doteq 1,6674$$

Pro $Q_2,\ Q_3,\ Q_4$ je postup výpočtů stejný.

$$Q_2 = 2,8202,\ Q_3 = 3,6285,\ Q_4 = 2,8341$$

Celková hodnota kvadraturní formule je rovna

$$\mathbf{Q} = Q_1 + Q_2 + Q_3 + Q_4 = 10,9502$$

V následující tabulce můžeme porovnat hodnoty jednoduché kvadraturní formule a složené tříbodové kvadraturní formule, kde m je počet uzlů jednoduché formule a M je počet podintervalů složené tříbodové formule. Obě formule jsou doplněny o skutečnou chybu.

m	M	GL formule	slož. GL formule	Skut. ch. GLF	Skut. ch. SGLF
3	1	10,9484	10,9484	$1{,}7677 \cdot 10^{-3}$	$1{,}7677 \cdot 10^{-3}$
6	2	10,9502	10,9501	$6{,}5156 \cdot 10^{-10}$	$3{,}8181 \cdot 10^{-5}$
9	3	10,9502	10,9502	$1{,}0481 \cdot 10^{-13}$	$3{,}5062 \cdot 10^{-6}$

Srovnání se složeným Simpsonovým pravidlem

Simpsonovo pravidlo patří mezi Newton-Cotesovy formule s ekvidistantními uzly. V těchto uzlech je funkce f proložena parabolou, je proto mnohem jednodušší na ruční výpočet, než složená Gaussova kvadratura.

Složené Simpsonovo pravidlo je tvaru

$$\int_a^b f(x)\ \mathrm{d}x = \frac{h}{3}\left[f(a) + 2\sum_{j=1}^{M-1} f(x_{2j}) + 4\sum_{j=1}^{M} f(x_{2j-1}) + f(b)\right]. \tag{4.3}$$

Konstanta **M** je počet subintervalů, z ní pak vychází hodnota kroku $h = \frac{(b-a)}{2M}$.

Příklad 4.5. *Porovnejme složené Simpsonovo pravidlo a složenou Gaussovu kvadraturu na příkladě 4.1 pro $M = 2$, tedy pro dva podintervaly.*

Hodnota složené Gaussovy kvadratury, pro dva podintervaly, je $Q = 10,9501$ s chybou $R(f) = 3,8181 \cdot 10^{-5}$.

Pro Simpsonovo pravidlo je nutné nejdříve najít hodnotu kroku a následně všechny uzly. Jelikož hodnota kroku je $h = \frac{(3-1)}{4} = \frac{2}{4} = \frac{1}{2}$, můžeme najít jednotlivé uzly formule

$$a = 1,\ x_1 = \frac{3}{2},\ x_2 = 2,\ x_3 = \frac{5}{2},\ b = 3.$$

Integrovaná funkce je $f(x) = e^x sin\ x$, podle vzorce (4.3) můžeme sestavit formuli

$$\int_1^3 e^x sin\ x \mathrm{d}x = \frac{1}{6}\left[e^1 sin\ 1 + 2e^2 sin\ 2 + 4\left(e^{\frac{1}{2}} sin\ \frac{1}{2} + e^{\frac{5}{2}} sin\ \frac{5}{2}\right) + \right.$$

$$\left. + e^3 sin\ 3\right] = 10,9342$$

V následující tabulce můžeme porovnat hodnoty složeného Simpsonova pravidla, složené Gauss-Legenrovy tříbodové formule a skutečnou chybu pro obě metody.

M	Simpson	Gauss	Skutečná chyba Simpson	Skutečná chyba Gauss
1	10,6657	10,9484	$2,8443 \cdot 10^{-1}$	$1,7677 \cdot 10^{-3}$
2	10,9342	10,9501	$1,6019 \cdot 10^{-2}$	$3,8181 \cdot 10^{-5}$
3	10,9471	10,9502	$3,0778 \cdot 10^{-3}$	$3,5062 \cdot 10^{-6}$
4	10,9492	10,9502	$9,6379 \cdot 10^{-4}$	$6,3333 \cdot 10^{-7}$
5	10,9498	10,9502	$3,9283 \cdot 10^{-4}$	$1,6714 \cdot 10^{-7}$

4.2. Gauss-Čebyševovy kvadraturní formule

Pro numerický výpočet integrálu s vahovou funkcí $w(x) = \frac{1}{\sqrt{1-x^2}}$ na intervalu $[-1, 1]$ se jedná o Gauss-Čebyševovy kvadraturní formule. Tyto formule využívají k výpočtům Čebyševovy ortogonální polynomy. Formule je tvaru

$$\int_{-1}^{1} \frac{f(x)}{\sqrt{1-x^2}}\, \mathrm{d}x = \sum_{i=0}^{n} A_i f(x_i) + R(f),$$

kde x_i je i-tý kořen Čebyševova ortogonálního polynomu n-tého stupně.
V případě Gauss-Čebyševovy formule je možné použít při odhadu chyby speciální vzorec [2, str.398].

$$R(f) = \frac{2\pi}{2^{2n+2}(2n+2)!} f^{(2n+2)}(\xi), \ \ |\xi| < 1 \tag{4.4}$$

U této formule jsou všechny koeficienty A_i stejné, zavisí pouze na hodnotě n. Výpočet je možný pomocí vzorce [2, str.398]

$$A = \frac{\pi}{n+1}$$

Pro výpočet hodnoty kvadraturní formule jsem vytvořil funkci
GCKF - **G**auss-**Č**ebyševova **K**vadraturní **F**ormule.
Vstupem je konstanta **m**, která udává počet kořenů, případně stupeň polynomu, a integrovaná funkce **f**.

```
function int=GCKF(m,f)
%Vstupy
%m - počet kořenů nebo-li stupeň polynomu
%f - integrovaná funkce
%Výstup
%int - hodnota kvadraturní formule
[x,~]=KCP(m);
int=sum(f(x))*(pi/(m));
```

Příklad 4.6. *Užitím Gauss-Čebyševovy kvadraturní formule pro $m = 3$, aproximujte integrál*

$$\int_{-1}^{1} \frac{cos\ x}{\sqrt{1-x^2}} \mathrm{d}x$$

Váhová funkce je rovna $w(x) = \frac{1}{\sqrt{1-x^2}}$ a z toho vyplývá $f(x) = cos\ x$. Jelikož $m = 3$ je třeba tří uzlů (kořenů) výpočtených z Čebyševova ortogonálního polynomu třetího stupně:

$$T_3(x) = 4x^3 - 3x$$

a kořeny jsou rovny

$$x_0 = 0,\ x_1 = \tfrac{\sqrt{3}}{2},\ x_2 = -\tfrac{\sqrt{3}}{2}$$

Následně je nutné vypočítat koeficienty kvadraturní formule. U této formule jsou všechny koeficienty stejné a výpočet přes vzorec je velmi jednoduchý.

$$A = \frac{\pi}{m} = \frac{\pi}{3}$$

Teď je možné sestavit kvadraturní formuli

$$\int_{-1}^{1} \frac{cosx}{\sqrt{1-x^2}} \mathrm{d}x \approx A(f(x_0) + f(x_1) + f(x_2)) = 2,4041$$

Následně je třeba určit chybu, u tohoto integrálu můžeme chybu pouze odhadnou pomocí vzorce(4.4), jelikož není možné najít primitivní funkci. V následující tabulce je uvedena hodnota integrálu spočítaného numericky a odhad chyby pro $m = 1, \ldots, 5$.

m	Hodnota integrálu	Odhad chyby
1	3,1416	$7{,}8540 \cdot 10^{-1}$
2	2,3884	$1{,}6362 \cdot 10^{-2}$
3	2,4041	$1{,}3635 \cdot 10^{-4}$
4	2,4039	$6{,}0872 \cdot 10^{-7}$
5	2,4039	$1{,}6909 \cdot 10^{-9}$

4.3. Gauss-Laguerrovy kvadraturní formule

V případě nekonečného intervalu $[0, \infty)$ je možné použít Gauss-Laguerrovy kvadraturní formule. Tyto formule využívají k výpočtům Laguerrovy ortogonální polynomy s vahou $w(x) = x^\alpha e^{-x}$, $\alpha > -1$. Formule je tvaru

$$\int_0^\infty x^\alpha e^{-x} f(x)\, \mathrm{d}x = \sum_{i=0}^{n} A_i f(x_i) + R(f)$$

kde x_i je i-tý kořen Laguerrova ortogonálního polynomu n-tého stupně.
V případě Gauss-Laguerrovy formule pro $\alpha = 0$ je možné použít při odhadu chyby speciální vzorec. [2, str.392]:

$$R(f) = \frac{((n+1)!)^2}{(2(n+1))!} f^{(2n+2)}(\gamma),\ 0 < \gamma < \infty. \tag{4.5}$$

Koeficienty této formule je možné spočítat integrací interpolačního polynomu(3.2) nebo pomocí speciálně odvozeného vzorce [5]

$$A_i = \frac{\Gamma(n+\alpha+2)x_i}{(n+1)!(n+2)^2[L_{n+2}^\alpha(x_i)]^2},\ i = 0, 1, \ldots, n,$$

kde L_{n+2}^α je Laguerrův ortogonální polynom stupně $n+2$ pro danou hodnotu α.

Pro výpočet koeficientů v MATLABU jsem vytvořil funkci
KGLa - **K**oeficienty **G**auss-**La**guerrovy formule
Vstupem je konstanta **m**, která udává počet kořenů, případně stupeň polynomu, a konstanta **a**, což je hodnota konstanty α. Výstupem je sloupcový vektor **A** koeficientů Kvadraturní formule.

```
function A=KGLa(m,a)
%Vstupy
%m - počet koeficientů případně počet kořenů
%a - hodnota konstanty alfa
%Výstup
```

```
%A - sloupcový vektor koeficientů kvadraturní formule
i=1;
[~,LaP]=KLaP(m+1,a);
x=roots(LaP(m+1,:));
A=zeros(m,1);
f=factorial(m);
while i<=m
g=(gamma(m+a+1)
A(i,1)=g*x(i,1))/(f*(m+1)^2*(polyval(LaP(m+2,:),x(i,1)))^2);
i=i+1;
end
```

Pro výpočet hodnoty kvadraturní formule jsem vytvořil funkci
GLaKF - **G**auss-**La**guerrova **K**vadraturní **F**ormule.
Vstupem je konstanta **m**, která udává počet kořenů, případně stupeň polynomu, konstanta **a**, hodnota konstanty α, a integrovaná funkce **f**.

```
function int=GLaKF(m,a,f)
%Vstupy
%m - počet kořenů nebo-li stupeň poylnomu
%a - hodnota konstanty alfa
%f - integrovaná funkce
%Výstup
%int - hodnota kvadraturní formule
if a>-1
[x,~]=KLaP(m,a);
A=KGLa(m,a);
int=sum(f(x).*A);
else
string("Parametr alfa musí být větší než -1")
end
```

Příklad 4.7. *Užitím Gauss-Laguerrovy kvadraturní formule pro m=3, aproximujte integrál*

$$\int_0^\infty e^{-x} sin\ x\ \mathrm{d}x$$

Hned na začátku je třeba vzít na vědomí vahovou funkci, která je zde rovna $w(x) = e^{-x}$ z toho vyplývá, že $f(x) = sin\ x$ a $\alpha = 0$. Jelikož $m = 3$, je tedy třeba tří uzlů(kořenů). Pro výpočet uzlů musíme nejdřív odvodit Legendrův polynom třetího stupně pro $\alpha = 0$

$$L_3 = -\frac{1}{6}x^3 + \frac{3}{2}x^2 - 3x + 1$$

a následně spočítáme kořeny polynomu třetího stupně.

$$x_0 = 6,2899,\ \ x_1 = 2,2943,\ \ x_2 = 0,4158$$

Dále je třeba spočítat koeficienty kvadraturní formule pomocí odvozeného vzorce, pro vzorec je třeba spočítat Laguerrův polynom čtvrtého stupně.

$$L_4 = -\frac{1}{6}x^3 + \frac{3}{2}x^2 - 3x + 1$$

$$A_0 = \frac{\Gamma(4)x_0}{3!(4)^2[L_4(x_0)]^2} = 0,0104$$

$$A_1 = \frac{\Gamma(4)x_1}{3!(4)^2[L_4(x_1)]^2} = 0,2785$$

$$A_2 = \frac{\Gamma(4)x_2}{3!(4)^2[L_4(x_2)]^2} = 0,7111$$

Přibližná hodnota integrálu pro $m = 3$ je tedy rovna

$$\int_0^\infty e^{-x} sin\ x\ \mathrm{d}x \approx A_0 f(x_0) + A_1 f(x_{1)}) + A_2 f(x_2) \doteq 0,4960$$

Následně je třeba určit chybu, u tohoto integrálu můžeme vypočítat jak odhad chyby tak i skutečnou chybu. Pro výpočet skutečné chyby je nutné znát přesnou hodnotu integrálu. Dvojitým použitím metody per partes dostaneme

$$\int_0^\infty e^{-x} sin\ x\ \mathrm{d}x = \lim_{t\to\infty} \left[-e^{-x} \frac{sin\ x + cos\ x}{2} \right]_0^t = \frac{1}{2}$$

Skutečnou chybu určíme jednoduchým rozdílem podle vztahu(1.2). V následující tabulce je uvedena hodnota integrálu spočítaného numericky, skutečná chyba a odhad chyby pro $m = 1, \dots, 5$

m	Hodnota integrálu	Skutečná chyba	Odhad chyby
1	0,8415	$3{,}4147 \cdot 10^{-1}$	$5{,}0000 \cdot 10^{-1}$
2	0,4325	$6{,}7541 \cdot 10^{-2}$	$1{,}6667 \cdot 10^{-1}$
3	0,4960	$3{,}9702 \cdot 10^{-3}$	$5{,}0000 \cdot 10^{-2}$
4	0,5049	$4{,}8793 \cdot 10^{-3}$	$1{,}4286 \cdot 10^{-2}$
5	0,4989	$1{,}0967 \cdot 10^{-3}$	$3{,}9683 \cdot 10^{-3}$

4.4. Gauss-Hermitovy kvadraturní formule

V případě, že je váhová funkce rovna $w(x) = e^{-x^2}$ na nekonečném intervalu $(-\infty, \infty)$ jde o Gauss-Hermitovy kvadraturní formule. Formule je tvaru

$$\int_{-\infty}^{\infty} e^{-x^2} f(x) = \sum_{i=0}^{n} A_i f(x_i) + R(f)$$

kde x_i je i-tý kořen Hermitova ortogonálního polynomu n-tého stupně.
V případě Gauss-Hermitovy fomule je možné použít při odhadu chyby speciální vzorec. [2, str.395].

$$R(f) = \frac{(n+1)!\sqrt{\pi}}{2^{n+1}(2n+2)!} f^{(2n+2)}(\xi)$$

Koeficienty této formule je možné spočítat integrací interpolačního polynomu(3.2) nebo pomocí speciálně odvozeného vzorce.

$$A_i = \frac{2^{n+2}(n+1)!\sqrt{\pi}}{[H_{n+2}(x_i)]^2}. \tag{4.6}$$

Kde H_{n+2} je Hermitův ortogonální polynom stupně $n+2$.
Pro výpočet koeficientů v MATLABU jsem vytvořil funkci
KGH - **K**oeficienty **G**auss-**H**ermitovy formule.
Vstupem je konstanta **m**, která udává počet kořenů, případně stupeň polynomu.
Výstupem je sloupcový vektor **A** koeficientů kvadraturní formule.

```
function A=KGH(m)
%Vstup
%m - počet kořenů, případně stupeň polynomu
%Výstup
%A - sloupcový vektor koeficientů kvadraturní formule
[~,HP]=KHP(m+1);
x=roots(HP(m+1,:));
i=1;
A=zeros(m,1);
f=factorial(m);
while i<=m
A(i,1)=(f*sqrt(pi)*(2^(m+1)))/((polyval(HP(m+2,:),x(i,1)))^2);
i=i+1;
end
```

Pro výpočet kvadraturní formule jsem vytvořil funkci **GHKF** - **G**auss-**H**ermitova **K**vadraturní **F**ormule. Vstupem je konstanta **m**, která udává počet kořenů, případně stupeň polynomu, a integrovaná funkce **f**.

```
function int=GHKF(m,f)
%Vstupy
%m - počet kořenů, případně stupeň polynomu
%f - integrovaná funkce
%Výstupy
%int - hodnota kvadraturní formule
[x,~]=KHP(m);
A=KGH(m);
int=sum(f(x).*A);
```

Příklad 4.8. *Užitím Gauss-Hermitovy kvadraturní formule pro m=4, aproximujte integrál*

$$\int_{-\infty}^{\infty} e^{-x^2} x \; sin \; x^2 \; \mathrm{d}x.$$

Hned na začátku je třeba vzít na vědomí vahovou funkci, která je zde rovna $w(x) = e^{-x^2}$ z toho vyplývá, že $f(x) = x\ sin\ x^2$. Jelikož $m = 4$, je třeba čtyř uzlů(kořenů). Pro výpočet uzlů musíme nejdřív spočítat Hermitův polynom čtvrtého stupně

$$H_4 = 16x^4 - 48x^2 + 12,$$

a následně spočítáme kořeny tohoto polynomu.

$$x_0 = -1,6507,\ x_1 = 1,6507,\ x_2 = -0,5246,\ x_3 = 0,5246$$

Dále je třeba spočítat koeficienty kvadraturní formule pomocí odvozeného vzorce, pro vzorec je třeba spočítat Hermitův polynom pátého stupně.

$$H_5 = 32x^5 - 160x^3 + 120x$$

$$A_0 = \frac{2^5 4! \sqrt{\pi}}{[H_5(x_0)]^2} = 0,0813$$

$$A_1 = \frac{2^5 4! \sqrt{\pi}}{[H_5(x_1)]^2} = 0,0813$$

$$A_2 = \frac{2^5 4! \sqrt{\pi}}{[H_5(x_2)]^2} = 0,8049$$

$$A_3 = \frac{2^5 4! \sqrt{\pi}}{[H_5(x_3)]^2} = 0,8049$$

Přibližná hodnota integrálu pro $m = 4$ je tedy rovna

$$\int_{-\infty}^{\infty} e^{-x^2} x\ sin\ x^2\ \mathrm{d}x \approx A_0 f(x_0) + A_1 f(x_1) + A_2 f(x_2) + A_3 f(x_3) \doteq -0,1148$$

Následně je třeba určit chybu, u tohoto integrálu můžeme vypočítat jak odhad chyby, tak i skutečnou chybu. Pro výpočet skutečné chyby je nutné vypočítat integrál pomocí substituce a metody per partes.

$$\int_{-\infty}^{\infty} e^{-x^2} x\ sin\ x^2\ \mathrm{d}x = \left[\frac{-e^{-x^2}}{4}\right]_{-\infty}^{\infty} = 0$$

Tento integrál je nutné spočítat jako nevlastní, tedy pomocí limit. Jelikož hodnota integrálu je rovna 0, chyba této formule je přímo rovna absolutní hodnotě kvadraturní formule.

Příklad 4.9. *Užitím Gauss-Hermitovy kvadraturní formule pro m=5, aproximujte integrál*

$$\int_{-\infty}^{\infty} e^{-x^2} cos\ x \ \mathrm{d}x$$

Váhová funkce je zde rovna $w(x) = e^{-x^2}$ z toho vyplývá, že $f(x) = cos\ x$. Jelikož $m = 5$, je třeba pěti uzlů(kořenů). Pro výpočet uzlů musíme nejdříve spočítat Hermitův polynom pátého stupně

$$H_5 = 32x^5 - 160x^3 + 120x$$

a následně spočítáme kořeny tohoto polynomu.

$$x_0 = 0,\ x_1 = -2,0202,\ x_2 = 2,0202,\ x_3 = -0,9586,\ x_4 = 0,9586$$

Dále je třeba spočítat koeficienty kvadraturní formule pomocí odvozeného vzorce, pro vzorec je třeba spočítat Hermitův polynom šestého stupně.

$$H_6 = 64x^6 - 480x^4 + 720x^2 - 120$$

$$A_0 = \frac{2^6 5! \sqrt{\pi}}{[H_6(x_0)]^2} = 0,9453$$

$$A_1 = \frac{2^6 5! \sqrt{\pi}}{[H_6(x_1)]^2} = 0,0200$$

$$A_2 = \frac{2^6 5! \sqrt{\pi}}{[H_6(x_2)]^2} = 0,0200$$

$$A_3 = \frac{2^6 5! \sqrt{\pi}}{[H_6(x_3)]^2} = 0,3936$$

$$A_4 = \frac{2^6 5! \sqrt{\pi}}{[H_6(x_4)]^2} = 0,3936$$

Přibližná hodnota integrálu pro $m = 5$ je tedy rovna

$$\int_{-\infty}^{\infty} e^{-x^2} x \, sin \, x^2 \, \mathrm{d}x \approx A_0 f(x_0) + A_1 f(x_1) + A_2 f(x_2) + A_3 f(x_3) + A_4 f(x_4) \doteq 1,3804$$

Následně je třeba určit chybu, jelikož není možné vypočítat přesnou hodnotu integrálu můžeme chybu pouze odhadnout. Pro odhad chyby použijeme vzorec (4.6). V následující tabulce je uvedena hodnota integrálu spočítaného numericky a odhad chyby pro $m = 1, \ldots, 5$

m	Hodnota integrálu	Odhad chyby
1	1,7725	$4{,}4311 \cdot 10^{-1}$
2	1,3475	$3{,}6923 \cdot 10^{-2}$
3	1,3820	$1{,}8463 \cdot 10^{-3}$
4	1,3803	$6{,}5940 \cdot 10^{-5}$
5	1,3804	$1{,}8317 \cdot 10^{-6}$

Závěr

Cílem práce bylo nastudovat Gaussovy kvadraturní formule a sestavit kódy v MATLABU. Gaussovy kvadraturní formule dosahují nejvyššího stupně přesnosti, dosahují tedy při daném počtu uzlů přesnějších výsledků než např. Newton-Cotesovy formule.

Narozdíl od Newton-Cotesových formulí, se kterými jsem se setkal v základním kurzu numerických metod, s Gaussovými formulemi jsem se seznámil teprve díky této práci.

Pro pochopení Gaussových kvadratur bylo třeba projít si obecnou kvadraturní formuli a tvar ortogonálních polynomů, bez nichž by nebylo možné dosáhnout té nejvyšší přesnosti. Tuto skutečnou jsem ověřil při porovnání jednoduché i složené Gauss-Legendrovy formule a složeného Simpsonova pravidla, kdy Gauss-Legendrova formule počítala s mnohem menší chybou.

Pro výpočet ortogonálních polynomů, jejich uzlů, koeficientů kvadraturní formule a celkové hodnoty kvadraturní formule jsem vytvořil kódy v MATLABU, které jsou k dispozici na přiloženém CD.

Jak je ukázáno na příkladech, dává nám Gaussova kvadratura univerzálnější přístup k výpočtům díky vahovým funkcím, do kterých můžeme začlenit například singularitu. A zároveň nám, díky konvergenci k přesné hodnotě, dává pro většinu funkcí spolehlivý způsob určení co nejpřesnější hodnoty.

Literatura

[1] HOROVÁ , Ivana, ZELINKA, Jiří: *Numerické metody.* Grafex, Blansko, 2004.

[2] HILDEBRAND, F., B.: *Introduction to NUMERICAL ANALYSIS.* Dover Publications, Mineola, 1987.

[3] RALSTON, A.: *Základy numerické matematiky.* ACADEMIA, Praha, 1978.

[4] Wikipedia – Domovská stránka [online]. [cit. 2018-27-03]. Dostupné z: https://en.wikipedia.org/wiki/Gaussian_quadrature.

[5] WolframMathWorld – Domovská stránka [online]. [cit. 2018-16-05]. Dostupné z: http://mathworld.wolfram.com/Laguerre-GaussQuadrature.html