MASARYKOVA UNIVERZITA
PŘÍRODOVĚDECKÁ FAKULTA
ÚSTAV MATEMATIKY A STATISTIKY

Bakalářská práce

BRNO 2014 **PETRA KOŠČÁKOVÁ**

MASARYKOVA UNIVERZITA
PŘÍRODOVĚDECKÁ FAKULTA
ÚSTAV MATEMATIKY A STATISTIKY

Ortogonální polynomy

Bakalářská práce

Petra Koščáková

Vedoucí práce: Mgr. Jiří Zelinka, Dr. **Brno 2014**

Bibliografický záznam

Autor:	Petra Koščáková Přírodovědecká fakulta, Masarykova univerzita Ústav matematiky a statistiky
Název práce:	Ortogonální polynomy
Studijní program:	Matematika
Studijní obor:	Obecná matematika
Vedoucí práce:	Mgr. Jiří Zelinka, Dr.
Akademický rok:	2013/2014
Počet stran:	ix + 45
Klíčová slova:	ortogonální polynomy; Hermitovy polynomy; Laguerrovy polynomy; Jacobiho polynomy; Legendrovy polynomy; Gegenbauerovy polynomy; Čebyševovy polynomy 1. typu; Čebyševovy polynomy 2. typu; Gaussovy kvadraturní formule; symetrické diferenciální operátory

Bibliographic Entry

Author:	Petra Koščáková Faculty of Science, Masaryk University Department of Mathematics and Statistics
Title of Thesis:	Orthogonal polynomials
Degree Programme:	Mathematics
Field of Study:	Mathematics
Supervisor:	Mgr. Jiří Zelinka, Dr.
Academic Year:	2013/2014
Number of Pages:	$\mathrm{ix} + 45$
Keywords:	orthogonal polynomials; Hermite polynomials; Laguerre polynomials; Jacobi polynomials; Legendre polynomials; Gegenbauer polynomials; Chebyshev polynomials of the first kind; Chebyshev polynomials of the second kind; Gaussian quadrature formula; symmetrical differential operators

Abstrakt

Bakalářská práce se zabývá ortogonálními polynomy a jejich použitím. Nejdříve jsou představeny vlastnosti ortogonálních polynomů a systémů. Následně je proveden rozbor základních druhů a na závěr jsou uvedeny některé způsoby použití. Součástí práce jsou zdrojové kódy algoritmů v softwaru MATLAB pro výpočet polynomů zmiňovaných typů a vykreslení jejich grafu.

Abstract

The thesis deals with orthogonal polynomials and their usage. Foremost, properties of orthogonal polynomials and systems are presented. Subsequently, an analysis of basic types is performed and in conclusion some ways of usage are listed. The paper also contains the source code of algorithms in software MATLAB to calculate mentioned polynomial types and plot their graphs.

Masarykova univerzita

Přírodovědecká fakulta

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

Student: **Petra Koščáková**

Studijní program - obor: **Matematika – Obecná matematika**

Ředitel Ústavu matematiky a statistiky PřF MU Vám ve smyslu Studijního a zkušebního řádu MU určuje bakalářskou práci s tématem:

Ortogonální polynomy a jejich použití

Orthogonal polynomials and their usage

Oficiální zadání: V práci se zaměřte na přehled základních typů ortogonálních polynomů, jejich vlastnosti a použití. Součástí práce může být také vytvoření softwarového balíku pro práci s ortogonálními polynomy.

Doporučená literatura:

HOROVA, Ivana a Jiří ZELINKA. *Numerické metody*. 2. vyd. Brno: Masarykova univerzita v Brně, 2004. 294 s. 3871/Př-2/04-17/31. ISBN 80-210-3317-7.

SZEGÖ, Gabor. *Orthogonal polynomials*. Providence: American Mathematical Society, 1991. xiii, 432. ISBN 0-8218-1023-5.

GAUTSCHI, Walter. *Orthogonal Polynomials (in Matlab)*. 2012.

Vedoucí bakalářské práce: Mgr. Jiří Zelinka, Dr.

Datum zadání bakalářské práce: říjen 2013

V Brně dne 31. 10. 2013

v.z.

prof. RNDr. Jiří Rosický, DrSc.
ředitel Ústavu matematiky a statistiky

Souhlasím se zadáním (datum, podpis):

4.3.2014 Koščáková
student(ka)

vedoucí práce

Poděkování

Na tomto místě bych chtěla poděkovat zejména vedoucímu práce, Mgr. Jiřímu Zelinkovi, Dr., za veškerý věnovaný čas, trpělivost a cenné rady při psaní této práce. Také bych chtěla poděkovat doc. Ing. Josefu Daňkovi, Ph.D. za podnětné nápady a nový pohled na moji práci.

Prohlášení

Prohlašuji, že jsem svoji bakalářskou práci vypracovala samostatně s využitím informačních zdrojů, které jsou v práci citovány.

Brno 29. května 2014 ....

Petra Koščáková

Obsah

Úvod

Pojem ortogonálních polynomů poprvé zazněl na začátku 19. století, a to v práci Adrien-Marie Legendra. Postupem času byly popsány další typy ortogonálních polynomů a začátkem 20. století se tímto tématem začala zabývat širší vědecká společnost.

V současnosti se ortogonální polynomy nejčastěji užívají v souvislosti s odhadem hodnot funkcí. K tomuto účelu postačí pouze znalosti tabulkových hodnot jednotlivých typů ortogonálních polynomů.

Cílem práce je poskytnout čtenáři základní znalosti týkající se studia ortogonálních polynomů a ukázat základní možnosti jejich použití. Nejprve jsou uvedeny hlavní definice a věty platné obecně pro libovolné typy ortogonálních polynomů, případně pak také ortogonálních systémů. Poté jsou představeny nejznámější druhy ortogonálních polynomů, jejich vlastnosti a také grafy. Na závěr jsou ukázány nejčastější způsoby použití ortogonálních polynomů.

Dalším úkolem pak bylo vytvořit programy v softwaru MATLAB pro výpočet základních typů ortogonálních polynomů.

Přehled použitého značení

Pro snažší orientaci v textu zde čtenáři předkládáme přehled základního značení, které se v celé práci vyskytuje.

(a,b)	otevřený interval
$[a,b]$	uzavřený interval
$\mathbb{C}$	množina všech komplexních čísel
$\mathbb{R}$	množina všech reálných čísel
$\mathbb{Z}$	množina všech celých čísel
$\mathbb{N}$	množina všech přirozených čísel
$\mathbb{N}_0$	množina všech nezáporných celých čísel
$L(I)$	Lebesqueův prostor funkcí absolutně intergovatelných na intervalu I
l	množina všech konvergentních posloupností
Π_j	množina všech polynomů stupně nejvýše j
$\overline{\Pi}_j$	množina všech normovaných polynomů stupně j
$\Gamma(n)$	gamma funkce

Kapitola 1

Základní vlastnosti

Nejdříve si uvedeme základní definice, které je nutno znát pro práci s ortogonálními polynomy. V další části pak budeme uvažovat obecně ortogonální systémy, mezi které lze samozřejmě uvažovat i ortogonální polynomy, takže si uvedeme nejdůležitější vlastnosti i pro ně.

Všechny zde zmíněné informace byly čerpány z: [1], [2], [4].

Definice 1.1. Nechť $w(x)$ je integrovatelná a nezáporná funkce definovaná na intervalu $[a,b] \subseteq \mathbb{R}$ a $w(x) > 0$ skoro všude na $[a,b]$. Pak tuto funkci budeme nazývat vahovou funkcí.

Poznámka. Skalárním součinem funkcí f a g budeme rozumět

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

pro takové funkce f, g, pro které existuje konečný integrál

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

kde $w(x)$ je daná vahová funkce.

Definice 1.2. Funkce f, g jsou na intervalu $[a,b]$ ortogonální s vahou w, jestliže platí

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

Definice 1.3. Systém funkcí $\{\phi_k\}_{k=0}^{\infty}$ se nazývá ortogonální systém, jestliže $\langle \phi_m, \phi_n\rangle = 0$ pro každé $m,n \in \mathbb{N}_0, m \neq n$.
Navíc se nazývá ortonormální, jestliže platí $\langle \phi_n, \phi_n\rangle = 1$.

Věta 1.1. *Pro danou vahovou funkci $w(x)$ na intervalu $[a,b]$ existují polynomy $p_i \in \Pi_i$, kde $i = 0,1,2,\dots$, takové, že $\langle p_j, p_k\rangle = 0$ pro $j \neq k$.*

Důkaz. Polynomy lze sestrojit pomocí Grammova-Schmidtova ortogonalizačního procesu. Zvolíme $p_0(x) = 1$ a další členy vypočteme následovně

$$
\begin{aligned}
p_1(x) &= x - \frac{\langle x, p_0\rangle}{\langle p_0, p_0\rangle} p_0(x) \\
p_2(x) &= x^2 - \frac{\langle x^2, p_0\rangle}{\langle p_0, p_0\rangle} p_0(x) - \frac{\langle x^2, p_1\rangle}{\langle p_1, p_1\rangle} p_1(x) \\
&\vdots \\
p_n(x) &= x^n - \frac{\langle x^n, p_0\rangle}{\langle p_0, p_0\rangle} p_0(x) - \frac{\langle x^n, p_1\rangle}{\langle p_1, p_1\rangle} p_1(x) - \ldots - \frac{\langle x^n, p_{n-1}\rangle}{\langle p_{n-1}, p_{n-1}\rangle} p_{n-1}(x)
\end{aligned}
$$

□

Věta 1.2. *Polynomy $p_i(x)$ definované v předchozí větě jsou definovány jednoznačně, a to vztahy*

$$
\begin{aligned}
p_{-1}(x) &\equiv 0, \\
p_0(x) &\equiv 1, \\
p_{i+1}(x) &= (A_{i+1}x + A_i)\, p_i(x) + A_{i-1} p_{i-1}(x), \qquad \textit{pro } i \geq 0,
\end{aligned}
$$

Důkaz. Nejprve ukážeme, že

$$p_{i+1}(x) = (A_{i+1}x + A_i)\, p_i(x) + A_{i-1} p_{i-1}(x) + A_{i-2} p_{i-2}(x) + \cdots + A_0 p_0(x).$$

Toto vyplývá z

$$p_{i+1}(x) - A_{i+1} x p_i(x) = A_i p_i(x) + A_{i-1} p_{i-1}(x) + A_{i-2} p_{i-2}(x) + \cdots + A_0 p_0(x). \tag{1.0.1}$$

Na pravé straně rovnice máme polynom stupně nejvýše i. Na levé straně je koeficient A_{i+1} zvolen tak, aby se rovnal hodnotě vedoucího koeficientu polynomu $p_{i+1}(x)$, tzn. že na levé straně máme polynom stupně nejvýše i. Pokud tedy budeme postupně volit koeficienty A_j, $j \in \mathbb{N}_0$, $j \leq i$ tak, aby se polynom $A_j p_j(x)$ vždy odečetl od polynomu $p_{i+1}(x)$ a polynomu $p_{i+1}(x)$ se snížil stupeň, je rovnost ověřena.

Nyní je třeba dokázat, že koeficienty $A_{i-2}, A_{i-3}, \ldots, A_0$ jsou rovny nule. Vynásobíme rovnici (1.0.1) členem $w(x) \cdot 1$ a zintegrujeme od a do b, kde volba a, b závisí na dané vahové funkci $w(x)$.

$$
\begin{aligned}
\int_a^b [p_{i+1}(x) - A_{i+1} x p_i(x)] \cdot w(x) \cdot 1 \,\mathrm{d}x = \int_a^b [A_i p_i(x) + A_{i-1} p_{i-1}(x) + A_{i-2} p_{i-2}(x) + \cdots + \\
+ A_0 p_0(x)] \cdot w(x) \cdot 1 \,\mathrm{d}x.
\end{aligned}
$$

Tuto rovnici můžeme následně upravit

$$
\begin{aligned}
\int_a^b p_{i+1}(x) \cdot w(x) \,\mathrm{d}x - A_{i+1} \int_a^b x p_i(x) \cdot w(x) \,\mathrm{d}x = A_i \int_a^b p_i(x) \cdot w(x) \,\mathrm{d}x + \\
+ A_{i-1} \int_a^b p_{i-1}(x) \cdot w(x) \,\mathrm{d}x + \cdots + A_0 \int_a^b p_0(x) \cdot w(x) \,\mathrm{d}x.
\end{aligned}
$$

Jednotlivé členy odpovídají skalárním součinům:

$$\langle p_{i+1}(x),1\rangle - A_{i+1}\langle p_i(x),x\rangle = A_i\langle p_i(x),1\rangle + A_{i-1}\langle p_{i-1}(x),1\rangle + \cdots + A_0\langle p_0(x),1\rangle.$$

Ty jsou všechny nulové, jelikož polynomy jsou kolmé na 1, kromě součinu $\langle p_0(x),1\rangle = B_0$. Ten je jistě nenulový, jelikož polynom $p_0(x)$ má stupeň nula, tedy $p_0(x)$ a 1 jsou lineárně závislé. Jelikož je ale levá strana rovnice rovna nule a víme, že koeficient B_0 je nenulový, musí platit, že $A_0 = 0$.

Opakujeme stejný proces, tentokrát ovšem vynásobíme rovnici (1.0.1) členem $w(x)\cdot x$. Po úpravách dostáváme rovnici

$$\langle p_{i+1}(x),x\rangle - A_{i+1}\langle p_i(x),x^2\rangle = A_i\langle p_i(x),x\rangle + \cdots + A_1\langle p_1(x),x\rangle + A_0\langle p_0(x),x\rangle.$$

Tentokrát bude nenulový pouze člen $\langle p_1(x),x\rangle = B_1$, z čehož vyplývá, že také $A_1 = 0$.

Takto budeme pokračovat pro další $w(x)\cdot x^j$, $j = 2,3,4,\dots$, dokud na levé straně nedostaneme nenulový člen. To nastane až pro x^{i-1}.

$$\langle p_{i+1}(x),x^{i-1}\rangle - A_{i+1}\langle p_i(x),x\cdot x^{i-1}\rangle = A_i\langle p_i(x),x^{i-1}\rangle + A_{i-1}\langle p_{i-1}(x),x^{i-1}\rangle + \cdots + \\ + A_0\langle p_0(x),x^{i-1}\rangle.$$

Na levé straně rovnice je nenulový člen $\langle p_i(x),x\cdot x^{i-1}\rangle$, zároveň však na pravé straně rovnice je nenulový člen $\langle p_{i-1}(x),x^{i-1}\rangle$. Tzn. $A_{i+1},A_{i-1} \neq 0$. Také po vynásobení rovnice x^n dostaneme nenulovou pravou i levou část rovnice, tedy A_i je různé od nuly. Pokud dosadíme vypočtené členy A_j, $j = 0,1,\dots,i+1$ do původní rovnice, dostáváme

$$p_{i+1}(x) = (A_{i+1}x + A_i)\, p_i(x) + A_{i-1}p_{i-1}(x),$$

což je tvrzení, které jsme měli dokázat. □

Věta 1.3. *Polynomy $p_i(x)$ tvoří bázi prostoru Π_n, tzn. každý polynom $p(x) \in \Pi_n$ lze vyjádřit jako lineární kombinaci ortogonálních polynomů $p_i(x) \in \overline{\Pi}_i, i \leq n$.*

Důkaz. Dle důkazu věty 1.1 víme, že polynom $p_i(x)$ je stupně přesně i. Z toho vyplývá, že množina polynomů $p_i(x)$ je lineárně nezávislá. Tzn. že prostor Π_n je generován $n+1$ ortogonálními polynomy $p_i(x)$, kde $i = 0,1,2,\dots,n$. □

Věta 1.4. *Pro všechny polynomy $p(x) \in \Pi_{n-1}$ a ortogonální polynomy $p_n(x)$ stupně n platí*

$$\langle p(x), p_n(x)\rangle = 0.$$

Důkaz. Důkaz vyplývá z tvrzení předchozí věty. Polynom $p(x)$ můžeme vyjádřit jako lineární kombinaci ortogonálních polynomů stupně nejvýše $n-1$:

$$p(x) = a_1p_1(x) + a_2p_2(x) + a_3p_3(x) + \cdots + a_kp_k(x), \quad \text{kde } k \leq n-1.$$

Pak skalární součin $\langle p(x), p_n(x)\rangle$ můžeme vyjádřit jako:

$$\langle p(x), p_n(x)\rangle = \langle a_1p_1(x) + a_2p_2(x) + a_3p_3(x) + \cdots + a_kp_k(x), p_n(x)\rangle = \\ a_1\langle p_1(x), p_n(x)\rangle + a_2\langle p_2(x), p_n(x)\rangle + a_3\langle p_3(x), p_n(x)\rangle + \cdots + a_k\langle p_k(x), p_n(x)\rangle.$$

Jelikož se jedná o ortogonální polynomy a $k < n$, všechny dílčí skalární součiny musejí být nulové. □

Věta 1.5. *Nechť $\{p_i(x)\}_{i=0}^{\infty}$ je systém ortogonálních polynomů s vahou $w(x)$ na intervalu $[a,b]$. Pak platí, že každý polynom $p_i(x)$ má všechny kořeny různé, reálné a všechny náleží do intervalu (a,b).*

Důkaz. Viz [1], strana 44, Theorem 3.3.1. □

Věta 1.6. *Nechť $\{p_i(x)\}_{i=0}^{\infty}$ je systém ortogonálních polynomů s vahou $w(x)$ na intervalu $[a,b]$. Potom:*

(a) $p_i'(x)p_{i-1}(x) - p_{i-1}'p_i(x) > 0$ *pro* $\forall x \in \mathbb{R}$, $i \geq 1$.

(b) Nechť pro kořeny $x_1, x_2, \ldots, x_n$ polynomu p_n platí $x_1 < x_2 < \cdots < x_n$ a $x_0 = a$, $x_{n+1} = b$. Potom každý interval $[x_k, x_{k+1}]$, kde $k = 0, 1, \ldots, n$ obsahuje právě jeden kořen polynomu p_{n+1}.

Důkaz. (a) Viz [1] strana 45, 3.3.6.

(b) Z (a) vyplývá, že polynomy p_n a p_{n-1} nemají společné kořeny.

□

Věta 1.7. *Mezi každými dvěma kořeny polynomu $p_n(x)$ se nachází alespoň jeden kořen polynomu $p_m(x)$, $m > n$.*

Důkaz. Viz [1] strana 46, Theorem 3.3.3. □

Kapitola 2

Vybrané druhy ortogonálních polynomů

Podle typů vahových funkcí rozpoznáváme různé druhy ortogonálních polynomů, z nichž nejčastěji užívané si ukážeme v této kapitole. Pro každý druh si zde uvedeme také nejdůležitější vlastnosti. Podrobnější informace lze nalézt v: [1] [2] [3] [4].

Mezi základní druhy patří Hermitovy polynomy, Laquerrovy polynomy a Jacobiho polynomy.

2.1 Hermitovy polynomy

Hermitovy polynomy jsou ortogonální s vahovou funkcí $w(x) = e^{-x^2}$ na intervalu $(-\infty, \infty)$.

Pro jejich výpočet platí rekurentní vztah:

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

Polynomy $H_0(x)$ a $H_1(x)$ jsou tvaru:

$$H_0(x) = 1, \quad H_1(x) = 2x.$$

Pro skalární součin platí:

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

Hermitovy polynomy jsou řešením diferenciální rovnice:

$$y'' - 2xy' + 2ny = 0.$$

Hodnotám Hermitových polynomů v bodě 0, tj. $H_n(0)$, se říká Hermitova čísla. Jsou dána rekurentním vztahem

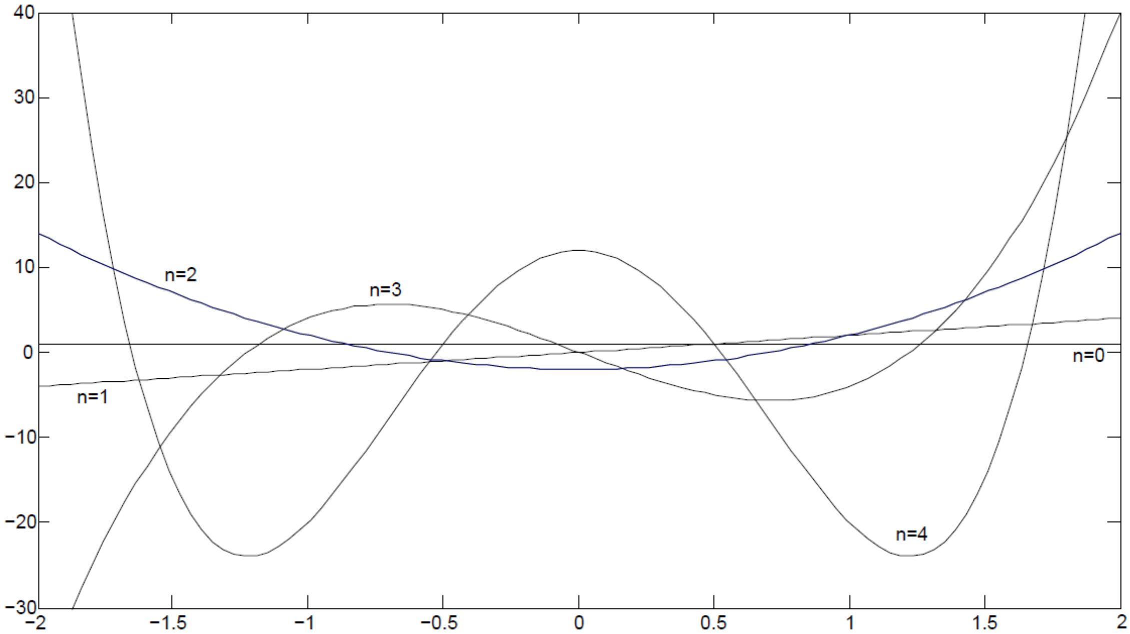
$$H_n = -2(n-1)H_{n-2}.$$

Jelikož $H_1(0) = 0$, je zřejmé, že všechna Hermitova čísla lichého řádu budou nulová. Platí tedy

$$H_n = \begin{cases} 0, & \text{pokud } n \text{ je liché,} \\ (-1)^{\frac{n}{2}} 2^{\frac{n}{2}} (n-1)!!, & \text{pokud } n \text{ je sudé,} \end{cases}$$

kde $(n-1)!! = 1 \cdot 3 \cdot 5 \cdots (n-1)$.

Následující graf znázorňuje průběh Hermitovy polynomy stupně $0, 1, 2, 3, 4$ na intervalu $[-2, 2]$.



Obrázek 2.1: Graf Hermitových polynomů stupně n

2.2 Laguerrovy polynomy

Laquerrovy polynomy jsou ortogonální s vahovou funkcí $w(x) = x^{\alpha}e^{-x}$, kde $\alpha > -1$, na intervalu $[0, \infty)$.

Lze je vyjádřit pomocí rekurentního vztahu:

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

Polynomy $L_0(x,\alpha)$ a $L_1(x,\alpha)$ jsou tvaru:

$$L_0(x,\alpha) = 1, \quad L_1(x,\alpha) = -x+\alpha+1.$$

Pro skalární součin platí vztah:

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

Laquerrovy polynomy jsou řešením diferenciální rovnice:

$$xy'' + (\alpha - x + 1)y' + ny = 0.$$

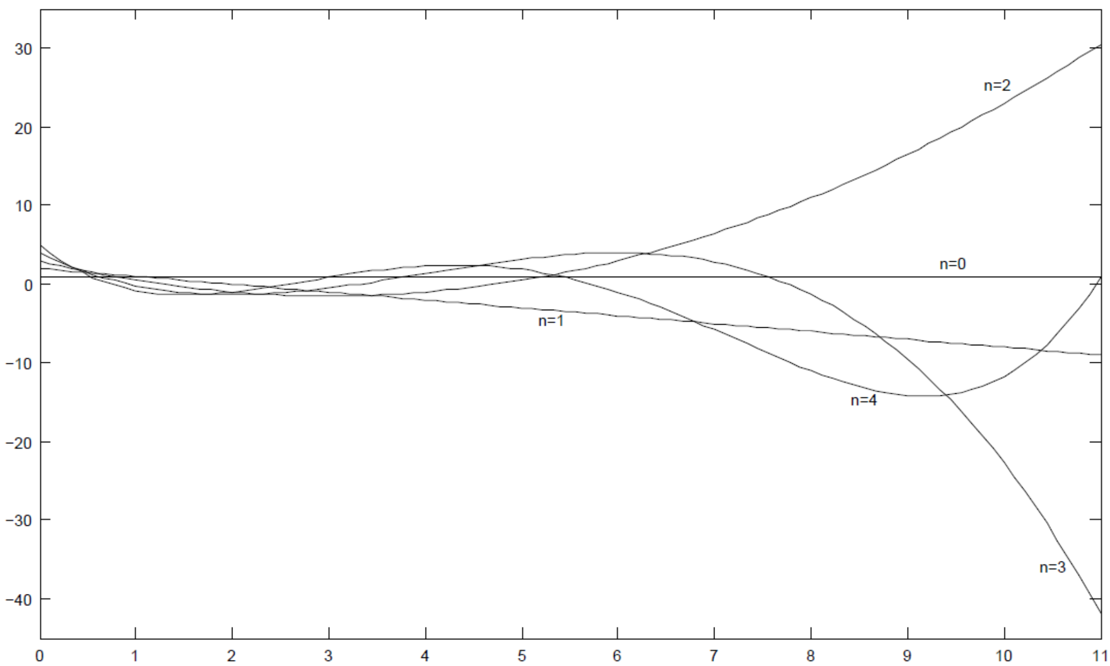
Mezi Hermitovými a Laguerrovými polynomy existují vztahy, které umožňují převod Hermitových polynomů na polynomy Laquerrovy, a to následujícím způsobem:

$$H_{2n}(x) = (-1)^n 2^{2n} n! \cdot L_n(x^2, -\frac{1}{2}), \quad H_{2n+1}(x) = (-1)^n 2^{2n+1} n! \cdot x \cdot L_n(x^2, \frac{1}{2}).$$

Pro Laguerrovy polynomy pak také platí:

- $$L_n(0,\alpha) = \frac{(\alpha+1)(\alpha+2)\cdots(\alpha+n)}{n!},$$
- $$L_n(x,\alpha) = L_n(x,\alpha+1) - L_{n-1}(x,\alpha).$$

Na následujícím grafu je znázorněn průběh Laguerrových polynomů stupně $0,1,2,3,4$ pro volbu parametru $\alpha = 1$ na intervalu $[0,11]$.



Obrázek 2.2: Graf Laguerrových polynomů stupně n

2.3 Jacobiho polynomy

Třetím základním typem ortogonálních polynomů jsou Jacobiho polynomy, které jsou ortogonální s vahovou funkcí $w(x) = (1-x)^\alpha(1+x)^\beta$, kde $\alpha,\beta > -1$, a to na intervalu $[-1,1]$.

Pro jejich výpočet platí rekurentní vztah:

$$\begin{aligned}P_{n+1}(x,\alpha,\beta) = {}& \frac{(2n+\alpha+\beta+1)[\alpha^2-\beta^2+(2n+\alpha+\beta)(2n+\alpha+\beta+2)x]}{2(n+1)(n+\alpha+\beta+1)(2n+\alpha+\beta)} P_n(x,\alpha,\beta) \\ & - \frac{2(n+\alpha)(n+\beta)(2n+\alpha+\beta+2)}{2(n+1)(n+\alpha+\beta+1)(2n+\alpha+\beta)} P_{n-1}(x,\alpha,\beta), \qquad n = 1,2,3,\ldots\end{aligned}$$

Polynomy $P_0(x,\alpha,\beta)$ a $P_1(x,\alpha,\beta)$ jsou tvaru:

$$P_0(x,\alpha,\beta) = 1, \quad P_1(x,\alpha,\beta) = \frac{1}{2}[2(\alpha+1)+(\alpha+\beta+2)(x-1)].$$

Pro skalární součin platí:

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

Jacobiho polynomy jsou řešením diferenciální rovnice:

$$(1-x^2)y'' + [\beta-\alpha-(\alpha+\beta+2)x]y' + n(n+\alpha+\beta+1)y = 0.$$

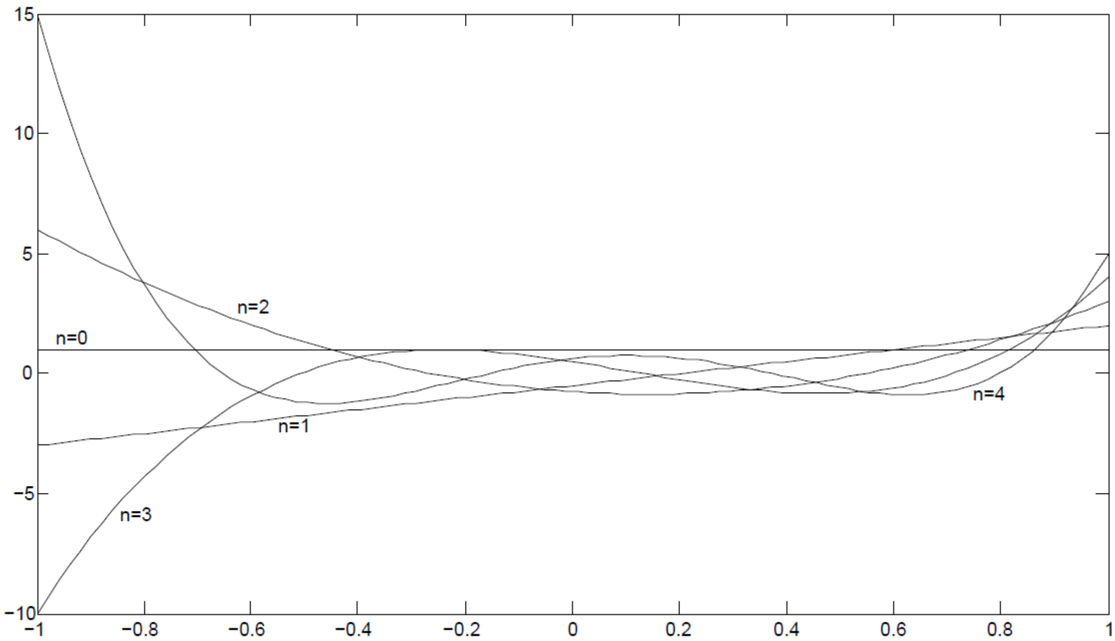
Pro Jacobiho polynomy platí symetrický vztah

$$P_n(x,\alpha,\beta) = (-1)^n P_n(x,\beta,\alpha).$$

Pro součet koeficientů Jacobiho polynomů platí:

$$P_n(1,\alpha,\beta) = \binom{n+\alpha}{n}.$$

Na následujícím grafu jsou zobrazeny Jacobiho polynomy stupně $0,1,2,3,4$ pro volbu parametrů $\alpha = 1$, $\beta = 2$ na intervalu $[-1,1]$.



Obrázek 2.3: Graf Jacobiho polynomů stupně n

Pro některé konkrétní volby parametrů α a β rozeznáváme další známé typy ortogonálních polynomů.

2.4 Legendrovy polynomy

Jde o typ Jacobiho polynomů pro volbu parametrů $\alpha, \beta = 0$. Jsou ortogonální s vahou $w(x) \equiv 1$ na intervalu $[-1,1]$.

Lze je vypočíst z rekurentního vztahu:

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

Polynomy $P_0(x)$ a $P_1(x)$ jsou tvaru:

$$P_0(x) = 1, \quad P_1(x) = x.$$

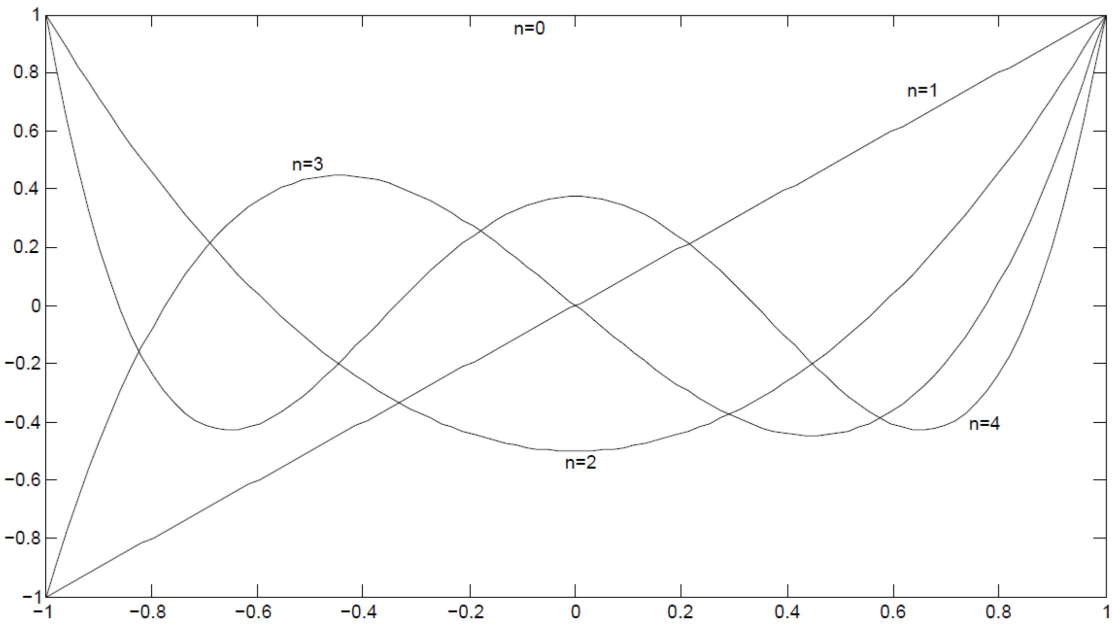
Pro skalární součin platí:

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

Legendrovy polynomy jsou řešením diferenciální rovnice:

$$(1-x^2)y'' - 2xy' + n(n+1)y = 0.$$

Následující graf zobrazuje průběh Legendrových polynomů stupně $0,1,2,3,4$ na intervalu $[-1,1]$.



Obrázek 2.4: Graf Legendrových polynomů stupně n

2.5 Gegenbauerovy polynomy

Gegenbauerovy, někdy též uváděny jako ultrasférické, polynomy jsou typem Jacobiho polynomů pro volbu parametrů $\alpha = \beta = \gamma - \frac{1}{2}$. Pro zjednodušení budeme místo γ používat označení α. Podmínkou je, že $\alpha > -\frac{1}{2}$. Jsou ortogonální s vahová funkcí $w(x) = (1 - x^2)^{\alpha - \frac{1}{2}}$ na intervalu $[-1, 1]$.

Pro jejich výpočet platí rekurentní vztah:

$$C_{n+1}(x,\alpha) = \frac{2(n+\alpha)}{n+1} x C_n(x,\alpha) - \frac{n+2\alpha-1}{n+1} C_{n-1}(x,\alpha), \qquad n = 1, 2, 3, \ldots$$

Polynomy $C_0(x,\alpha)$ a $C_1(x,\alpha)$ jsou tvaru:

$$C_0(x,\alpha) = 1, \quad C_1(x,\alpha) = 2\alpha x.$$

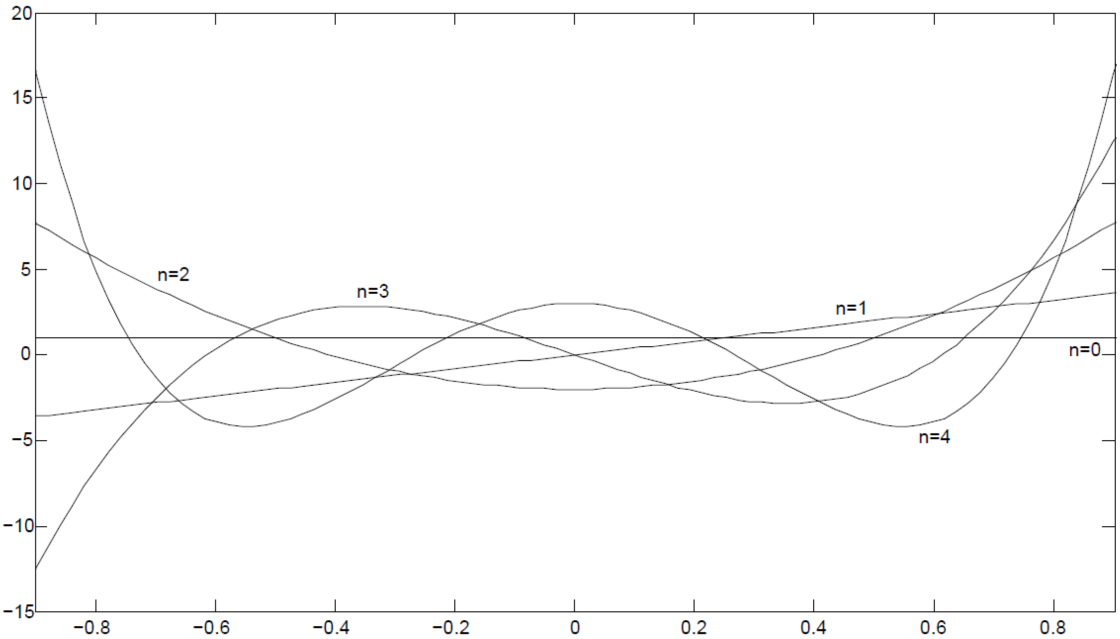
Pro skalární součin platí:

$$\int_{-1}^{1} (1-x^2)^{\alpha-\frac{1}{2}} C_m(x,\alpha) C_n(x,\alpha)\,\mathrm{d}x = \begin{cases} 0, & \text{pro } m \neq n; \\ \pi 2^{1-2\alpha} \frac{\Gamma(n+2\alpha)}{(n+\alpha)\Gamma^2(n)\Gamma(n+1)}, & \text{pro } m = n. \end{cases}$$

Gegenbauerovy polynomy jsou řešením diferenciální rovnice:

$$(1-x^2)y'' - (2\alpha+1)xy' + n(n+2\alpha)y = 0.$$

Na následujícím grafu je znázorněn průběh Gegenbauerových polynomů stupně $0, 1, 2$, 3, 4 pro volbu parametru $\alpha = 2$ na intervalu $[-1, 1]$.



Obrázek 2.5: Graf Gegenbauerových polynomů

Speciálním případem jsou Gegenbauerovy polynomy pro volbu parametru $\alpha = 0$. V daném případě jsou totiž polynomy stupně $n \geq 1$ rovny nule. Tomuto případu ovšem odpovídá následující druh polynomů.

2.6 Čebyševovy polynomy 1. typu

Vahová funkce je $w(x) = (1-x^2)^{-\frac{1}{2}} = \frac{1}{\sqrt{1-x^2}}$ a polynomy jsou ortogonální na intervalu $[-1,1]$.

Lze je vypočíst pomocí rekurentního vztahu:

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

Polynomy $T_0(x)$ a $T_1(x)$ jsou tvaru:

$$T_0(x) = 1, \quad T_1(x) = x.$$

Pro skalární součin platí:

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

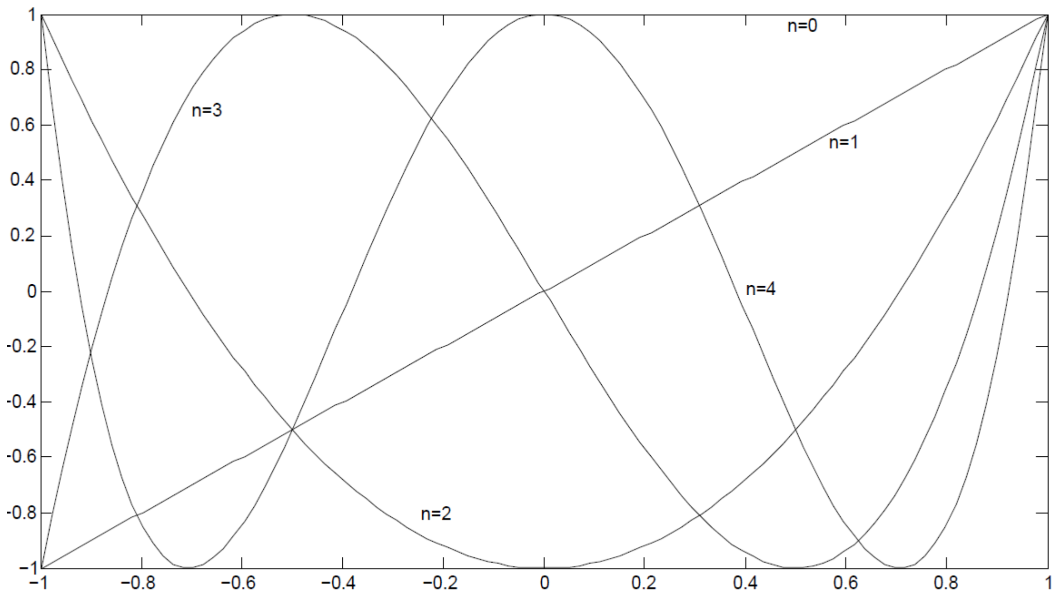
Tyto polynomy splňují diferenciální rovnici:

$$(1-x^2)y'' - xy' + n^2y = 0.$$

Pro Čebyševovy polynomy platí:

$$T_n(x) = cos(n \cdot arccos(x)).$$

V následující grafu je zobrazen průběh Čebyševových polynomů 1. typu stupně $n = 0,1,2,3,4$ na intervalu $[-1,1]$.



Obrázek 2.6: Graf Čebyševových polynomů 1. typu stupně n

2.7 Čebyševovy polynomy 2. typu

I v tomto případě se jedná o speciální typ Gegenbauerových polynomů, ovšem tentokrát pro volbu parametru $\alpha = 1$. Polynomy jsou ortogonální s vahou $w(x) = (1-x^2)^{\frac{1}{2}}$ na intervalu $[-1,1]$.

Pro výpočet platí rekurentní vztah:

$$U_{n+1}(x) = 2xU_n(x) - U_{n-1}(x), \qquad n = 1,2,3,\ldots$$

Polynomy $U_0(x)$ a $U_1(x)$ jsou tvaru:

$$U_0(x) = 1, \quad U_1(x) = 2x.$$

Pro skalární součin platí:

$$\int_{-1}^{1} (1-x^2)^{\frac{1}{2}} U_m(x) U_n(x)\,\mathrm{d}x = \begin{cases} 0, & \text{pro } m \neq n; \\ \frac{\pi}{2}, & \text{pro } m = n. \end{cases}$$

Polynomy jsou řešením diferenciální rovnice:

$$(1-x^2)y'' - 3xy' + n(n+2)y = 0.$$

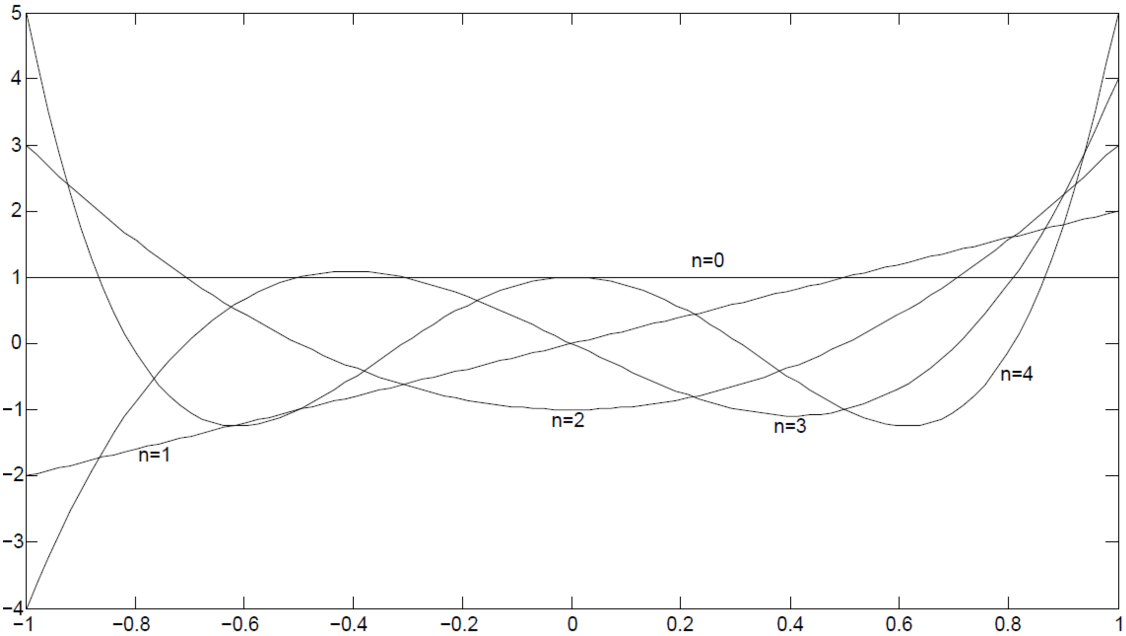
Pro Čebyševovými polynomy 2. typu platí:

$$2T_m(x)T_n(x) = T_{m+n}(x) + T_{m-n}(x), \quad m \geq n.$$

Mezi Čebyševovými polynomy prvního a druhého typu platí několik vztahů:

$$\begin{aligned} T_n(x) &= U_n(x) - xU_{n-1}(x), \\ T_n(x) &= xU_{n-1} - U_{n-2}(x), \\ T_n(x) &= \frac{1}{2}(U_n(x) - U_{n-2}(x)), \\ (1-x^2)U_n(x) &= xT_n(x) - T_{n+1}(x), \\ 2(x^2-1)U_{m-1}(x)U_{n-1}(x) &= T_{m+n}(x) - T_{m-n}(x), \quad m \geq n. \end{aligned}$$

Následující graf znázorňuje průběh Čebyševových polynomů 2. typu stupně $0,1,2,3,4$ na intervalu $[-1,1]$.



Obrázek 2.7: Graf Čebyševových polynomů 2. typu stupně *n*

2.8 Tvorba grafů a programů

V této kapitole byly pro ilustraci použity použity grafy vytvořené pomocí programu MATLAB. [8], [9]

Zároveň byl pro každý výše zmiňovaný typ polynomů vytvořen v MATLABu program. Samotný program slouží k výpočtu daného polynomu stupně zvoleného uživatelem. V případě polynomů, které vyžadují zadání dalšího parametru, uživatel kromě stupně zvolí také tento parametr. Program zkontroluje, zda zvolený parametr vyhovuje omezujícím podmínkám a v opačném případě zobrazí chybové hlášení.

Každý program má počet vstupních proměnných odpovídající počtu parametrů v rekurentní rovnici, tzn. jednu až tři.

Výstupní proměnné jsou pak vždy dvě. Do první proměnné se uloží požadovaný polynom ve tvaru, který používá program MATLAB. Do druhé proměnné se pak uloží polynom v symbolickém tvaru.

Dále byly vytvořeny programy, které kromě výpočtu zobrazí graf daného polynomu.

Zdrojové kódy těchto programů jsou uvedeny v příloze práce.

Kapitola 3

Použití ortogonálních polynomů

3.1 Aproximace funkcí

V následující části si ukážeme, jak lze použít ortogonální polynomy pro aproximaci funkcí. Motivací je, že s některými složitými funkcemi se obtížně provádějí výpočty. Proto si ukážeme možnost, jak funkci nahradit lineární kombinací ortogonálních polynomů, které lze snadno vypočíst, a s tímto odhadem původní funkce se nám poté bude snadněji pracovat.

Všechna tvrzení budeme uvádět pro libovolné ortogonální systémy, tedy nejen pro polynomy. [5]

V této části budeme uvažovat funkce s komplexním oborem hodnot definované na obecném intervalu $I = (a,b)$.

Prostor $L^2(I)$ je Lebesgueův prostor se skalárním součinem

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

obsahující funkce, pro které platí

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

Tento skalární součin indukuje normu

$$\|f(x)\| = \left(\int_a^b |f(x)|^2\,\mathrm{d}x\right)^{\frac{1}{2}}$$

a zároveň také metriku

$$\rho(f,g) = \|f-g\| = \left(\int_a^b |f(x)-g(x)|^2\,\mathrm{d}x\right)^{\frac{1}{2}}.$$

Definice 3.1. Nechť $\{\phi_k\}_{k=0}^{\infty}$ je ortonormální systém. Čísla

$$c_k = \int_a^b f(x)\overline{\phi_k(x)}\,\mathrm{d}x$$

se nazývají Fourierovy koeficienty funkce $f(x)$ vzhledem k systému $\{\phi_k\}_{k=0}^{\infty}$.

Poznámka. Výběr ortonormálního systému ovlivní možnost volby intervalu $I = (a,b)$, na kterém je definován prostor $L^2(I)$, a to zejména pokud se jedná např. o systém ortonormálních polynomů, které nejsou ortogonální na celém $\mathbb{R}$.

Jelikož v následujícím textu nebudeme uvažovat žádný konkrétní interval (a,b), budeme pro zjednodušení dále uvádět místo $L^2(I)$ označení L^2.

Definice 3.2. Funkční řada
$$\sum_{k=0}^{\infty} c_k\phi_k$$
se nazývá Fourierova řada funkce $f(x)$ vzhledem k ortonormálnímu systému $\{\phi_k\}_{k=0}^{\infty}$.

Použitím částečných součtů Fourierovy řady definujeme funkce
$$f_n = \sum_{k=0}^{n} c_k\phi_k.$$
Pomocí posloupnosti těchto funkcí budeme chtít aproximovat danou funkci f. Ukážeme, za jakých podmínek posloupnost f_n konverguje k funkci f pro $n \to \infty$ (jelikož uvažujeme normovaný prostor, jedná se o konvergenci v normě).

Lemma 3.1. *Nechť $\{\phi_k\}_{k=0}^{\infty}$ je ortonormální systém a $f(x) \in L^2$. Pak pro libovolná komplexní čísla $a_1, a_2, \dots, a_n$ platí:*
$$\|f - \sum_{k=0}^{n} a_k\phi_k\| \geq \|f - f_n\|.$$
K rovnosti dojde pouze v případě, že $a_1 = c_1, a_2 = c_2, \dots, a_n = c_n$.

Důkaz.
$$\begin{aligned}
\|f - \sum_{k=0}^{n} a_k\phi_k\|^2 &= \langle f - \sum_{k=0}^{n} a_k\phi_k, f - \sum_{k=0}^{n} a_k\phi_k\rangle = \\
&= \langle f, f\rangle - \langle f, \sum_{k=0}^{n} a_k\phi_k\rangle - \langle \sum_{k=0}^{n} a_k\phi_k, f\rangle + \langle \sum_{k=0}^{n} a_k\phi_k, \sum_{k=0}^{n} a_k\phi_k\rangle = \\
&= \|f\|^2 - \sum_{k=0}^{n} c_k\bar{a}_k - \sum_{k=0}^{n} a_k\bar{c}_k + \sum_{k=0}^{n} a_k\bar{a}_k = \\
&= \|f\|^2 - \sum_{k=0}^{n} (\bar{a}_k c_k + a_k\bar{c}_k) + \sum_{k=0}^{n} a_k\bar{a}_k = \\
&= \|f\|^2 + \sum_{k=0}^{n} (a_k\bar{a}_k - a_k\bar{c}_k - \bar{a}_k c_k + c_k\bar{c}_k) - \sum_{k=0}^{n} a_k\bar{a}_k - \\
&- \sum_{k=0}^{n} c_k\bar{c}_k + \sum_{k=0}^{n} |a_k|^2 = \\
&= \|f\|^2 + \sum_{k=0}^{n} \|a_k - c_k\|^2 - \sum_{k=0}^{n} |a_k|^2 - \sum_{k=0}^{n} |c_k|^2 + \sum_{k=0}^{n} |a_k|^2 = \\
&= \|f\|^2 + \sum_{k=0}^{n} \|a_k - c_k\|^2 - \sum_{k=0}^{n} |c_k|^2.
\end{aligned}$$

Prostřední člen je nezáporný a nabývá nejmenší hodnoty, právě když $a_k = c_k$ pro $k = 1,2,\dots,n$, zároveň ostatní členy nezávisí na hodnotě a_k, z čehož vyplývá, že i celá pravá strana rovnice nabývá nejmenší hodnoty.

Pro $a_k = c_k$ dostáváme:

$$\|f\|^2 = \sum_{k=0}^{n} |c_k|^2 + \|f - \sum_{k=0}^{n} c_k\phi_k\|^2.$$

Pravá strana původní rovnice v druhé mocnině je pak rovna

$$\|f - f_n\|^2 = \langle f - \sum_{k=0}^{n} c_k\phi_k, f - \sum_{k=0}^{n} c_k\phi_k\rangle = \|f\|^2 - \sum_{k=0}^{n} |c_k|^2.$$

□

Poznámka. Z předchozí věty vyplývá, že

$$\|f\|^2 \geq \sum_{k=0}^{n} |c_k|^2.$$

Tedy řada

$$\sum_{k=0}^{\infty} |c_k|^2$$

má omezenou posloupnost částečných součtů, která je zjevně neklesající. Z toho vyplývá, že je řada konvergentní a platí pro ni Besselova nerovnost

$$\|f\|^2 \geq \sum_{k=0}^{\infty} |c_k|^2.$$

Zároveň lze taky vidět, že funkce f_n je nejlepší aproximací funkce f mezi všemi lineárními kombinacemi funkcí ϕ_0, ϕ_1, ϕ_2,..., ϕ_n.

Lemma 3.2. *Fourierova řada*

$$\sum_{k=0}^{\infty} c_k\phi_k$$

konverguje k funkci f v normě L^2, právě když platí Parsevalova rovnost

$$\sum_{k=0}^{\infty} |c_k|^2 = \|f\|^2.$$

Důkaz. Funkce f_n konvergují k funkci f v normě L^2, právě když

$$\|f - \sum_{k=0}^{n} c_k\phi_k\|^2 \to 0 \quad \text{pro } n \to \infty.$$

Z důkazu předešlého lemmatu víme, že

$$\|f - \sum_{k=0}^{n} c_k\phi_k\|^2 = \|f\|^2 - \sum_{k=0}^{n} |c_k|^2.$$

Proto je konvergence f_n k f ekvivalentní s rovností

$$\sum_{k=0}^{n} |c_k|^2 = \|f\|^2.$$

□

Lemma 3.3. *Nechť $f_n \to f$ a $g_n \to g$ pro $n \to \infty$ v L^2. Pak*

$$\langle f_n, g_n \rangle \to \langle f, g \rangle,$$

neboli skalární součin je spojitý funkcionál na L^2 v metrice jím indukované.

Důkaz.

$$\langle f_n, g_n \rangle = \langle f + f_n - f, g + g_n - g \rangle = \langle f, g \rangle + \langle f_n - f, g \rangle + \langle f, g_n - g \rangle + \langle f_n - f, g_n - g \rangle.$$

Jelikož v každém Hilbertově prostoru platí Cauchyho-Schwartzova nerovnost

$$|\langle f, g \rangle| \leq \|f\| \|g\|,$$

můžeme upravit pravou část rovnice. Protože pro $n \to \infty$

$$|\langle f_n - f, g \rangle| \leq \|f_n - f\| \|g\| \to 0,$$

víme, že

$$\langle f_n - f, g \rangle \to 0.$$

Stejně pak

$$\langle f, g_n - g \rangle \to 0, \quad \langle f_n - f, g_n - g \rangle \to 0.$$

Z toho vyplývá, že

$$\langle f_n, g_n \rangle \to \langle f, g \rangle.$$

□

Lemma 3.4. *Nechť pro $n \to \infty$*

$$f_n = \sum_{k=0}^{n} c_k \phi_k \to f.$$

Pak řada

$$\sum_{k=0}^{\infty} c_k \phi_k$$

je Fourierovou řadou funkce f a c_k jsou Fourierovy koeficienty funkce f.

Důkaz.

$$\langle f, \phi_k \rangle = \int_a^b f \bar{\phi}_k \, \mathrm{d}x = \lim_{n \to \infty} \int_a^b f_n \bar{\phi}_k \, \mathrm{d}x = \lim_{n \to \infty} c_k = c_k$$

podle předchozího lemmatu 3.3 . □

Věta 3.5. *(Rieszova-Fischerova) Nechť $\{c_k\}_{k=0}^{\infty}$ je taková posloupnost, že*

$$\sum_{k=0}^{\infty} |c_k|^2$$

konverguje. Pak existuje funkce, jejíž Fourierovy koeficienty jsou rovny c_k, a řada

$$\sum_{k=0}^{\infty} c_k \phi_k$$

konverguje k funkci f v normě L^2.

Důkaz. Jelikož je metrický prostor L^2 úplný, platí v něm Bolzano-Cauchyova podmínka, tj. posloupnost f_n konverguje, právě když pro $m, n \to \infty$ platí

$$\int_a^b |f_m - f_n|^2 \, \mathrm{d}x \to 0.$$

V tomto případě

$$f_m = \sum_{k=0}^{m} c_k \phi_k, \qquad f_n = \sum_{k=0}^{n} c_k \phi_k,$$

tedy pro $m, n \to \infty$, $m < n$ je

$$\int_a^b |f_m - f_n|^2 \, \mathrm{d}x = \sum_{k=m+1}^{n} |c_k|^2 \to 0,$$

protože číselná řada

$$\sum_{k=0}^{\infty} |c_k|^2$$

konverguje a splňuje tedy Bolzano-Cauchyovu podmínku. Víme tedy, že existuje funkce $f \in L^2$ taková, že $f_n \to f$ v L^2. Z předchozího lemmatu 3.4 pak víme, že koeficienty c_k jsou Fourierovy koeficienty funkce f. □

Definice 3.3. Ortonormální systém $\{\phi_k\}_{k=0}^{\infty}$ se nazývá úplný, jestliže když libovolná funkce f splňuje $\langle f, \phi_n \rangle = 0$ pro $n = 0, 1, \ldots$, pak platí $f = 0$.

Poznámka. Z definice tedy vyplývá, že k úplnému ortogonálnímu systému neexistuje nenulová funkce kolmá ke každému členu systému.

Věta 3.6. *Nechť $\{\phi_k\}_{k=0}^{\infty}$ je úplný ortonormální systém a c_k jsou Fourierovy koeficienty funkce f. Pak*

$$\sum_{k=0}^{\infty} c_k \phi_k \to f.$$

Důkaz. Z Besselovy nerovnosti vyplývá, že

$$\sum_{k=0}^{\infty} |c_k|^2 \leq \|f\|^2 < \infty.$$

Z úplnosti systému $\{\phi_k\}_{k=0}^{\infty}$ pak plyne, že koeficienty c_k určují funkci f jednoznačně. Funkce f tedy splňuje podmínky pro funkci z Rieszovy-Fischerovy věty 3.5 . □

Věta 3.7. *Nechť $f,\ g \in L^2$, c_k jsou Fourierovy koeficienty funkce f, d_k jsou Fourierovy koeficienty funkce g. Pak*

$$\int_a^b f\bar{g}\,\mathrm{d}x = \sum_{k=0}^{\infty} c_k \bar{d}_k.$$

Důkaz.

$$\int_a^b f_n\bar{g}\,\mathrm{d}x = \int_a^b \sum_{k=0}^{n} (c_k\phi_k)\,\bar{g}\,\mathrm{d}x = \int_a^b \sum_{k=0}^{n} (c_k\phi_k\bar{g})\,\mathrm{d}x = \sum_{k=0}^{n}\left(c_k\int_a^b \phi_k\bar{g}\,\mathrm{d}x\right) = \sum_{k=0}^{n} c_k\bar{d}_k.$$

Pro $n \to \infty$ je $f_n \to f$, z čehož vyplývá, že

$$\int_a^b f\bar{g}\,\mathrm{d}x = \sum_{k=0}^{\infty} c_k \bar{d}_k.$$

□

Poznámka. Z předchozí věty vyplývá, že zobrazení $f \to \{c_k\}_{k=0}^{\infty}$ z prostoru L^2 do l^2, kde l^2 je prostor posloupností $(x_i)_{i=0}^{\infty}$ komplexních čísel splňujících

$$\sum_{i=0}^{\infty} |x_i|^2 < \infty,$$

je izomorfismus, tzn. zachovává skalární součin.

Na závěr si ukážeme samotný výpočet odhadu funkce. Nejdříve si uvedeme příklad pro spojitou funkci, poté pro funkci nespojitou.

Příklad: Aproximujte funkci $f(x) = sin(\pi x)\cdot cos(2\pi x)$ pomocí Legendrových polynomů $(n = 3)$.

Řešení: Nejdříve vypočteme první čtyři Legendrovy polynomy.

$$P_0(x) = 1, \quad P_1(x) = x, \quad P_2(x) = \frac{3}{2}x^2 - \frac{1}{2}, \quad P_3(x) = \frac{5}{2}x^3 - \frac{3}{2}x.$$

Vypočteme Fourierovy koeficienty funkce $f(x)$:

$$c_0 = \frac{\int_{-1}^{1} sin(\pi x)\cdot cos(2\pi x)\cdot 1\,\mathrm{d}x}{\int_{-1}^{1}\cdot 1^2\,\mathrm{d}x} = 0,$$

$$c_1 = \frac{\int_{-1}^{1} sin(\pi x)\cdot cos(2\pi x)\cdot x\,\mathrm{d}x}{\int_{-1}^{1}\cdot x^2\,\mathrm{d}x} = -\frac{1}{\pi},$$

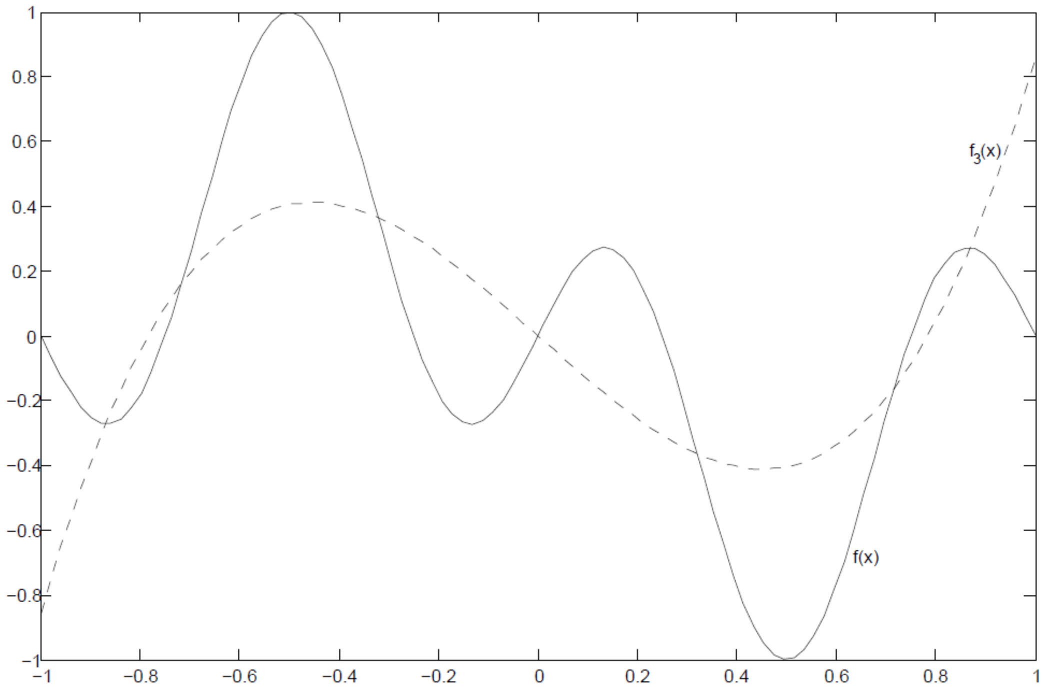
$$c_2 = \frac{\int_{-1}^{1} sin(\pi x)\cdot cos(2\pi x)\cdot\left(\frac{3}{2}x^2 - \frac{1}{2}\right)\mathrm{d}x}{\int_{-1}^{1}\cdot\left(\frac{3}{2}x^2 - \frac{1}{2}\right)^2\mathrm{d}x} = 0,$$

$$c_3 = \frac{\int_{-1}^{1} sin(\pi x)\cdot cos(2\pi x)\cdot\left(\frac{5}{2}x^3 - \frac{3}{2}x\right)\mathrm{d}x}{\int_{-1}^{1}\cdot\left(\frac{5}{2}x^3 - \frac{3}{2}x\right)^2\mathrm{d}x} = -\frac{21\pi^2 - 455}{9\pi^3}.$$

Funkce $f_3(x)$ je pak tvaru

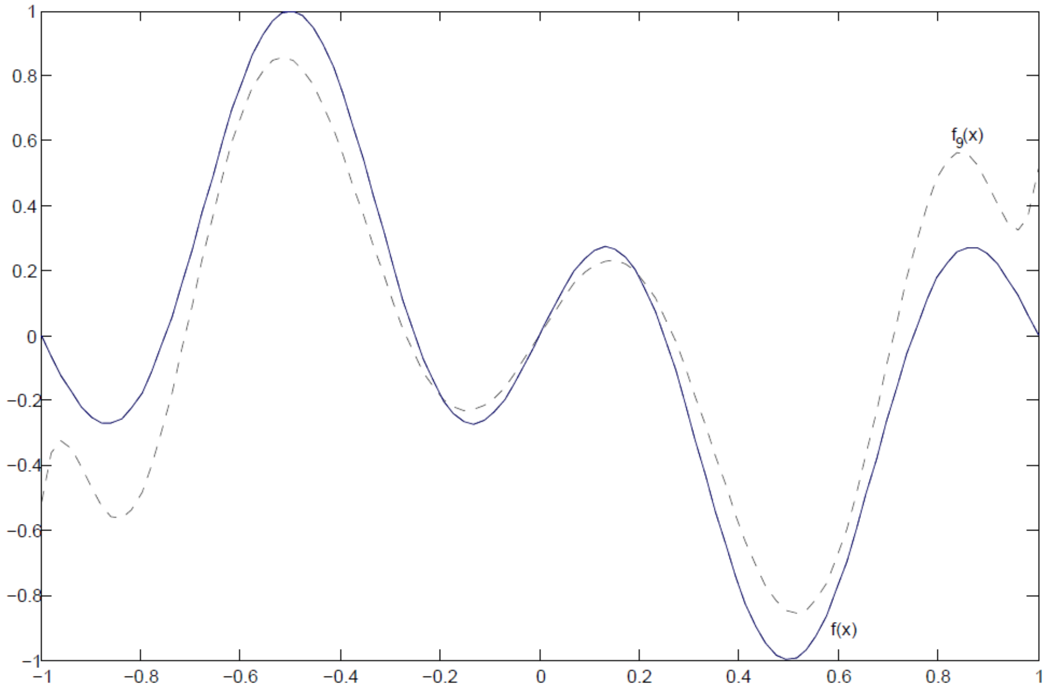
$$f_3(x) = -\frac{1}{\pi}\cdot x - \frac{21\pi^2 - 455}{9\pi^3}\cdot\left(\frac{5}{2}x^3 - \frac{3}{2}x\right) = -\frac{105\pi^2 - 2275}{18\pi^3}x^3 + \frac{63\pi^2 - 1383}{18\pi^3}x$$
$$= \frac{1}{18\pi^3}\cdot\left[-\left(105\pi^2 - 2275\right)x^3 + \left(63\pi^2 - 1383\right)x\right].$$

Pro ilustraci si ukažme graf původní funkce a vypočtené aproximace:



Obrázek 3.1: Graf funkce $f(x) = sin(\pi x)\cdot cos(2\pi x)$ a její aproximace $f_3(x)$

Z grafu lze vidět, že stupeň aproximace je nedostatečný. Ukažme si, jak by vypadal graf pro aproximaci vyššího stupně. Všechny sudé Fourierovy koeficienty budou rovny nule, proto zvolme například stupeň 9.



Obrázek 3.2: Graf funkce $f(x) = sin(\pi x) \cdot cos(2\pi x)$ a její aproximace $f_9(x)$

Z grafu lze usuzovat, že v tomto případě nám devátý stupeň poskytuje již dobrou aproximaci.

Příklad: Aproximujte funkci $f(x) = sgn(x)$ pomocí Čebyševových polynomů 2. typu ($n = 3$).

Řešení: Vypočteme příslušné Čebyševovy polynomy 2. stupně.

$$U_0(x) = 1, \quad U_1(x) = 2x, \quad U_2(x) = 4x^2 - 1, \quad U_3(x) = 8x^3 - 4x.$$

Nyní vyjádříme Fourierovy koeficienty funkce $f(x)$:

$$c_0 = \frac{\int_{-1}^{1}(1-x^2)^{\frac{1}{2}} sgn(x) \cdot 1 \, \mathrm{d}x}{\int_{-1}^{1}(1-x^2)^{\frac{1}{2}} \cdot 1^2 \, \mathrm{d}x} = 0,$$

$$c_1 = \frac{\int_{-1}^{1}(1-x^2)^{\frac{1}{2}} sgn(x) \cdot 2x \, \mathrm{d}x}{\int_{-1}^{1}(1-x^2)^{\frac{1}{2}} \cdot (2x)^2 \, \mathrm{d}x} = \frac{8}{3\pi},$$

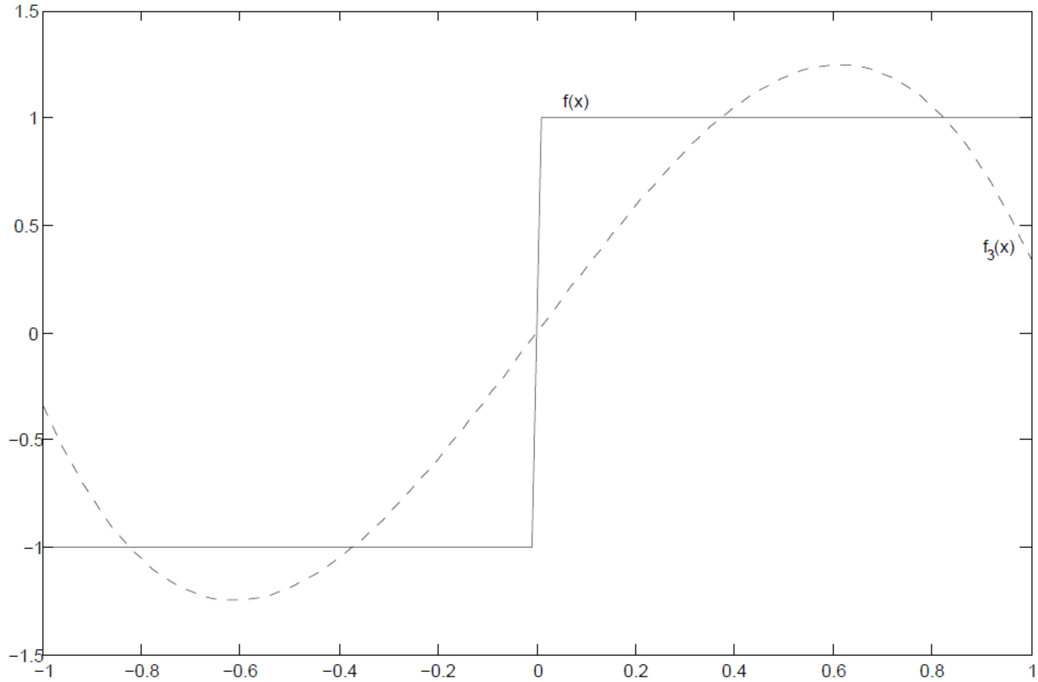
$$c_2 = \frac{\int_{-1}^{1}(1-x^2)^{\frac{1}{2}} sgn(x) \cdot (4x^2-1) \, \mathrm{d}x}{\int_{-1}^{1}(1-x^2)^{\frac{1}{2}} \cdot (4x^2-1)^2 \, \mathrm{d}x} = 0,$$

$$c_3 = \frac{\int_{-1}^{1}(1-x^2)^{\frac{1}{2}} sgn(x) \cdot (8x^3-4x) \, \mathrm{d}x}{\int_{-1}^{1}(1-x^2)^{\frac{1}{2}} \cdot (8x^3-4x)^2 \, \mathrm{d}x} = -\frac{16}{15\pi}.$$

Funkce $f_3(x)$ má tvar

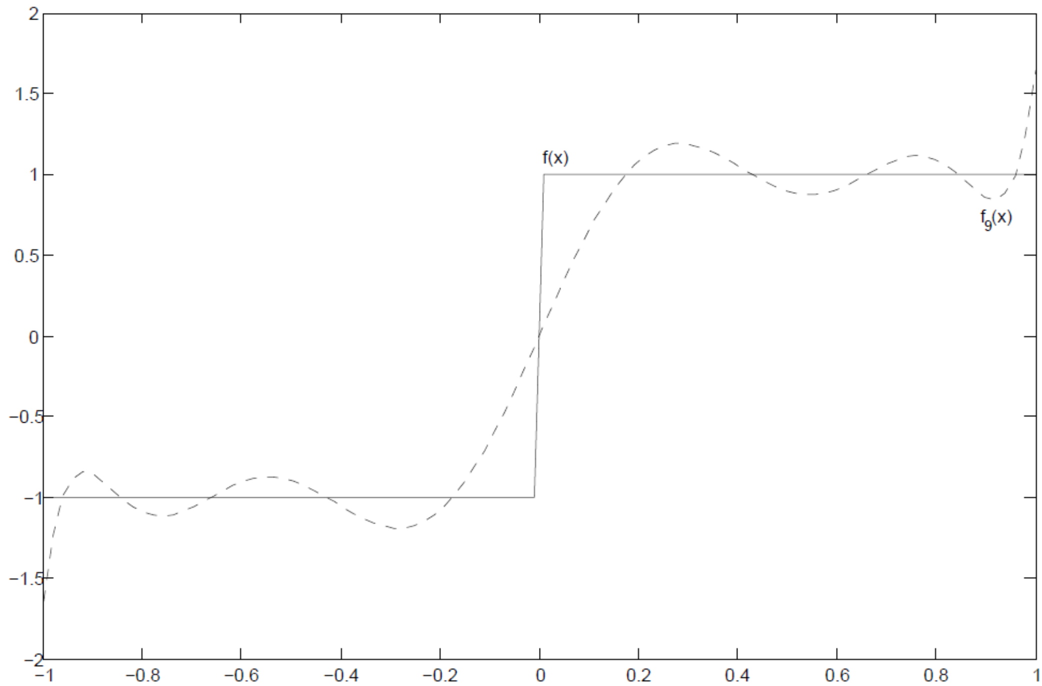
$$f_3(x) = \frac{8}{3\pi} \cdot 2x - \frac{16}{15\pi} \cdot (8x^3 - 4x) = \frac{144}{15\pi} x - \frac{128}{15\pi} x^3.$$

Graf původní a aproximované funkce vypadá následovně:



Obrázek 3.3: Graf funkce $f(x) = sgn(x)$ a její aproximace $f_3(x)$

Pro zajímavost si ukážeme ještě graf s aproximací 9. stupně. Z výpočtů se ukázalo, že nenulové jsou pouze liché koeficienty.



Obrázek 3.4: Graf funkce $f(x) = sgn(x)$ a její aproximace $f_9(x)$

3.2 Numerické integrování

Mějme integrál
$$I(f) = \int_a^b w(x) \cdot f(x) \, \mathrm{d}x. \tag{3.2.1}$$
Budeme chtít spočítat jeho hodnotu, což pokud jeho primitivní funkce nejde vyjádřit explicitně, nelze. Naším cílem bude daný integrál co nejlépe aproximovat a vypočíst hodnotu této aproximace.

Z konstrukce Riemannova integrálu, kdy jeho hodnotu chápeme jako obsah plochy mezi grafem funkce f a osou x a tuto plochu pokrýváme disjunktními obdélníky, se nabízí aproximace ve tvaru
$$I(f) \approx \sum_{i=0}^{n} A_i f(x_i),$$
kde body $x_i \in [a,b]$, $i = 0,1,\ldots,n$ a čísla $A_i \in \mathbb{R}$, $i = 0,1,\ldots,n$.

Definice 3.4. Výraz
$$Q(f) = \sum_{i=0}^{n} A_i f(x_i) \tag{3.2.2}$$
se nazývá kvadraturní formule, kde čísla $A_i \in \mathbb{R}$, $i = 0,1,\ldots,n$ jsou koeficienty kvadraturní formule a navzájem různé body $x_i \in [a,b]$, $i = 0,1,\ldots,n$ jsou uzly kvadraturní formule.

Definice 3.5. Rozdíl
$$R(f) = \int_a^b w(x) f(x) \, \mathrm{d}x - \sum_{i=0}^{n} A_i f(x_i)$$
se nazývá chyba kvadraturní formule.

Definice 3.6. Řekneme, že kvadraturní formule (3.2.2) má stupeň přesnosti N, jestliže
$$R(x^{(j)}) = 0, \quad j = 0,1,\ldots,N, \quad R(x^{(N+1)}) \neq 0.$$

Poznámka. Nejedná se o chybu kvadraturní formule, ale o algebraický stupeň přesnosti.

Věta 3.8. *Kvadraturní formule užívající $n+1$ uzlů má stupeň přesnosti nejvýše $2n+1$.*

Důkaz. Sporem. Nechť má kvadraturní formule (3.2.2) stupeň přesnosti $2n+2$ a nechť x_i, $i = 0,1,\ldots,n$ jsou uzly této kvadraturní formule (celkem tedy $n+1$ uzlů). Položme
$$\omega_{n+1}^2(x) = (x-x_0)^2 \cdot (x-x_1)^2 \cdots (x-x_n)^2$$
polynom stupně $2n+2$. Chyba kvadraturní formule pro výpočet integrálu
$$\int_a^b \omega_{n+1}^2(x) w(x) \, \mathrm{d}x$$
pak je
$$R(\omega_{n+1}^2(x)) = \int_a^b \omega_{n+1}^2(x) w(x) \, \mathrm{d}x - \sum_{i=0}^{n} A_i \omega_{n+1}^2(x_i) = \int_a^b \omega_{n+1}^2(x) w(x) \, \mathrm{d}x,$$

jelikož

$$\omega_{n+1}^2(x_i) = 0 \quad \text{pro } i = 0, 1, \ldots, n,$$

neboť x_i, $i = 0, 1, \ldots, n$ jsou kořeny polynomu ω_{n+1}. Z předpokladu, že kvadraturní formule má stupeň přesnosti $2n+2$ pak plyne, že

$$R(\omega_{n+1}^2(x)) = \int_a^b \omega_{n+1}^2(x) w(x)\,\mathrm{d}x = 0.$$

Toto ovšem nemůže nastat, jelikož $\omega_{n+1}^2(x)w(x)$ je zaručeně nezáporná funkce. Stupeň přesnosti kvadratické formule je tedy nejvýše $2n+1$. □

Při řešení úloh bývají často kladeny požadavky na výběr koeficientů a uzlů kvadraturní formule, např. předem zadané hodnoty některých uzlů či rovnost koeficientů kvadraturní formule. Pro naše potřeby nyní uvažujme případ, kdy nejsou zadány žádné omezující podmínky.

Nyní se budeme zabývat problémem, jak volit koeficienty a uzly pro dosažení maximálního stupně přesnosti kvadraturní formule, a zda je to vůbec možné.

Věta 3.9. *Nechť má kvadraturní formule* (3.2.2) *pro výpočet integrálu* (3.2.1) *stupeň přesnosti alespoň n. Nechť* $\{p_n\}_{n=0}^{\infty}$, $p_n \in \Pi_n$, *je ortogonální systém polynomů na intervalu* $[a,b]$ *s vahou w. Pak má tato formule stupeň přesnosti* $2n+1$ *právě tehdy, když uzly této kvadraturní formule jsou kořeny polynomu* $p_{n+1} \in \Pi_{n+1}$.

Poznámka. Kvadratickou formuli se stupněm přesnosti n lze vždy sestrojit, a to integrací interpolačního polynomu. Koeficienty lze vyjádřit takto:

$$A_i = \int_a^b w(x) l_i(x)\,\mathrm{d}x, \quad i = 0, 1, \ldots, n,$$

kde x_i, $i = 0, 1, \ldots, n$ jsou kořeny ortogonálního polynomu, který je tvaru

$$\omega_{n+1}(x) = (x - x_0)(x - x_1)\cdots(x - x_n)$$

a koeficienty $l_i(x)$, což jsou příslušné fundamentální polynomy v Lagrangeově interpolačním polynomu, lze pak spočítat jako

$$l_i(x) = \frac{\omega_{n+1}(x)}{\omega\prime_{n+1}(x_i)(x - x_i)}, \quad i = 0, 1, \ldots, n.$$

Viz [2], strana 228, Věta 9.2.

Důkaz. „⇒“: Nechť má kvadraturní formule $Q(f)$ stupeň přesnosti $2n+1$. Nechť polynom $\omega_{n+1}(x) \in \bar{\Pi}_n$ x_i, $i = 0, 1, 2, \ldots, n$ jsou jeho kořeny a zároveň uzly kvadraturní formule $Q(f)$. Z toho vyplývá, že také pro polynom $u_n\omega_{n+1}(x)$, kde $u_n \in \Pi_n$, je kvadraturní formule přesná. Proto tedy

$$\int_a^b \omega_{n+1}(x) u_n(x) w(x)\,\mathrm{d}x = \sum_{i=0}^{n} A_i \omega_{n+1}(x_i) u_n(x_i).$$

Jelikož ale $\omega_{n+1}(x_i) = 0$ pro všechna x_i, kde $i = 0, 1, \dots, n$, pravá strana rovnice je nulová a tedy

$$\int_a^b \omega_{n+1}(x) u_n(x) w(x)\,\mathrm{d}x = 0.$$

Víme, že polynom $\omega_{n+1}(x)$ je ortogonální ke všem polynomům třídy Π_n s vahou $w(x)$, z čehož vyplývá, že polynom $\omega_{n+1}(x)$ je totožný s polynomem $p_{n+1}(x) \in \Pi_n$.

„$\Leftarrow$“: Nechť jsou uzly kvadraturní formule $Q(f)$ kořeny polynomu $p_{n+1} \in \Pi_n$ ze systému ortogonálních polynomů $\{p_n\}$ s vahou $w(x)$ na intervalu (a, b). Všechny kořeny tohoto polynomu jsou reálné, navzájem různé a leží v intervalu (a, b) (viz Věta (1.5)). Nechť má tato formule stupeň přesnosti alespoň n (viz Poznámka).

Nechť $P_{2n+1}(x) \in \Pi_{2n+1}$ je libovolný polynom stupně $2n+1$. Pak lze tento polynom vyjádřit jako součet násobku polynomu $p_{n+1}(x)$ a zbytku:

$$P_{2n+1}(x) = p_{n+1}(x) u_n(x) + r_n(x),$$

kde polynomy $r_n(x), u_n(x) \in \Pi_n$. Nyní na polynom P_{2n+1} použijeme kvadraturní formuli a vyjádříme chybu aproximace integrálu

$$\begin{aligned} R(P_{2n+1}) &= \int_a^b \omega(x) P_{2n+1}(x)\,\mathrm{d}x - \sum_{i=0}^{n} A_i P_{2n+1}(x_i) = \\ &= [\int_a^b \omega(x) p_{n+1}(x) u_n(x)\,\mathrm{d}x - \sum_{i=0}^{n} A_i p_{n+1}(x_i) u_n(x_i) + \\ &+ \int_a^b \omega(x) r_n(x)\,\mathrm{d}x - \sum_{i=0}^{n} A_i r_n(x_i)] = 0, \end{aligned}$$

jelikož polynom $p_{n+1}(x)$ je ortogonální k polynomu $u_n(x)$ s vahou $w(x)$, dále pak polynom $p_{n+1}(x_i) = 0$ pro $i = 0, 1, \dots, n$, protože uzly x_i byly voleny jako kořeny polynomu $p_{n+1}(x)$. Druhá závorka je pak rovna nule, jelikož je kvadraturní formule aplikovaná na polynom $r_n(x)$ stupně nejvýše n a stupeň přesnosti této kvadraturní formule je alespoň n, tedy $R(r_n) = 0$ a pak i $R(P_{2n+1} = 0$. Jelikož polynom $P_{2n+1}(x) \in \Pi_{2n+1}$ byl libovolný polynom, stupeň přesnosti kvadraturní formule je $2n+1$. □

Definice 3.7. Nechť má kvadraturní formule (3.2.2) maximální stupeň přesnosti, tj. $2n+1$. Pak se tato formule nazývá Gaussova kvadraturní formule.

Věta 3.10. *Pro koeficienty Gaussovy kvadraturní formule platí:*

(a)

$$A_i > 0, \quad i = 0, 1, \dots, n,$$

(b)

$$\sum_{i=0}^{n} A_i = \int_a^b w(x)\,\mathrm{d}x.$$

Důkaz. (a) Gaussovy kvadraturní formule mají stupeň přesnosti $2n+1$ a jsou tedy přesné i pro polynomy stupně $2n$. Zvolme takovéto polynomy, např. kvadraturu fundamentálních polynomů v Lagrangeově interpolačním polynomu, tj.

$$l_j^2(x) = \left(\frac{p_{n+1}(x)}{p\prime_{n+1}(x_j)(x-x_j)} \right)^2, \quad j = 0, 1, \ldots, n.$$

Pro výpočet integrálu z těchto polynomů pak platí:

$$I(l_j^2(x)) = \int_a^b w(x) l_j^2(x)\,\mathrm{d}x = \sum_{i=0}^{n} A_i l_j^2(x_i).$$

Jelikož ale $l_j^2(x_i) = \delta_{ij}$, kde δ_{ij} značí Kroneckerovo delta, $i, j = 0, 1, \ldots, n$, pak

$$A_j = \int_a^b w(x) l_j^2(x)\,\mathrm{d}x > 0,$$

což je zaručeně kladné číslo.

(b) Mějme funkci $f(x) \equiv 1$. Pro tuto funkci je je Gaussova kvadraturní formule přesná a po její aplikaci dostáváme

$$\int_a^b w(x)\,\mathrm{d}x = \sum_{i=0}^{n} A_i.$$

□

Věta 3.11. *Nechť $f \in C^{(2n+2)}[a,b]$. Chyba Gaussovy kvadraturní formule má tvar*

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

kde $p_{n+1}(x) = \omega_{n+1}(x)$.

Důkaz. Viz [2], strana 234, Věta 9.6. □

Poznámka. Podle užívaných vahových funkcí rozlišujeme některé typy kvadraturních formulí. Mezi ty nejpoužívanější patří Gaussova-Legendreova kvadraturní formule s vahovou funkcí $w(x) = 1$ na intervalu $[-1,1]$, Gaussova-Čebyševova kvadraturní formule s vahovou funkcí $w(x) = \frac{1}{\sqrt{1-x^2}}$ na intervalu $[-1,1]$ a Gaussova-Hermitova kvadraturní formule s vahovou funkcí $w(x) = e^{-x^2}$ na intervalu $(-\infty,\infty)$.

Poznámka. Oproti ostatním typům kvadraturních formulí, které obecně toto nesplňují, mají Gaussovy formule jednu význačnou vlastnost, a to že s rostoucím počtem uzlů konverguje posloupnost Gaussových formulí k přesné hodnotě integrálu.

Příklad: Pomocí Gaussovy-Hermitovy kvadraturní formule (pro $n = 2$) vypočtěte integrál

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

Poznámka. Tento integrál lze vypočíst a jeho přesná hodnota je $\frac{\sqrt{\pi}}{\sqrt[4]{e}} \doteq 1,380388$.

Řešení: Přibližnou hodnotu integrálu si vyjádříme následovně:

$$\int_{-\infty}^{\infty} e^{-x^2} \cdot cos\,x\,\mathrm{d}x \approx \sum_{i=0}^{2} A_i f(x_i) = I.$$

Za uzly x_i budeme volit kořeny Hermitova polynomu, v našem případě 3. stupně. Ty jsou tři a mají hodnotu 0 a $\pm 1,224745$. Vahová funkce je tvaru

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

tedy funkce $f(x) = cos\,x$. Hodnotu koeficientů A_i pak vypočteme pomocí rovnic:

$$\begin{aligned} A_0 + A_1 + A_2 &= \int_{-\infty}^{\infty} 1 \cdot e^{-x^2}\,\mathrm{d}x = \sqrt{\pi}, \\ A_0x_0 + A_1x_1 + A_2x_2 &= \int_{-\infty}^{\infty} x \cdot e^{-x^2}\,\mathrm{d}x = 0, \\ A_0x_0^2 + A_1x_1^2 + A_2x_2^2 &= \int_{-\infty}^{\infty} x^2 \cdot e^{-x^2}\,\mathrm{d}x = \frac{\sqrt{\pi}}{2}. \end{aligned}$$

Dosadíme za $x_0 = -1,224745$, $x_1 = 0$ a $x_2 = 1,224745$ a vypočteme koeficienty A_0, A_1, A_2, čímž dostaneme, že $A_0 = 0,295409$, $A_1 = 1,181636$, $A_2 = 0,295409$. Dostáváme tedy rovnici

$$\begin{aligned} I &= 0,295409 \cdot cos(-1,224745) + 1,181636 \cdot cos\,0 + 0,295409 \cdot cos(1,224745) \\ &\doteq 1,382033. \end{aligned}$$

Pro jednotlivé typy vybraných ortogonálních polynomů lze v následujících tabulkách nalézt hodnoty kořenů a koeficientů A_i.

Gaussova-Hermitova kv. f.		
n	x_i	A_i
1	$\pm\frac{\sqrt{2}}{2} \doteq \pm 0,707107$	$0,886227$
2	0	$1,181636$
	$\pm\sqrt{\frac{3}{2}} \doteq \pm 1,224745$	$0,295409$
3	$\pm\sqrt{\frac{3}{2} - \sqrt{\frac{3}{2}}} \doteq \pm 0,524648$	$0,804914$
	$\pm\sqrt{\frac{3}{2} + \sqrt{\frac{3}{2}}} \doteq \pm 1,605680$	$0,081313$

Tabulka 3.1: Tabulka Gaussovy-Hemitovy kvadraturní formule stupně n

Gaussova-Laguerrova kv. f.		
n	x_i	A_i
1	$2-\sqrt{2} \doteq 0,585786$	$2,414214$
	$2+\sqrt{2} \doteq 3,414214$	$-0,414214$
2	$0,415775$	$2,675973$
	$2,294280$	$-0,785659$
	$6,289945$	$0,109686$
3	$0,322548$	$2,927017$
	$1,745761$	$-1,217722$
	$4,536620$	$0,318917$
	$9,395071$	$-0,028212$

Tabulka 3.2: Tabulka Gaussovy-Laguerrovy kvadraturní formule pro hodnotu parametru $\alpha = 0$ stupně n

Gaussova-Jacobiho kv. f.		
n	x_i	A_i
1	$\frac{1-2\sqrt{2}}{7} \doteq -0,261204$	$1,353553$
	$\frac{1+2\sqrt{2}}{7} \doteq 0,546918$	$0,646447$
2	$-0,507787$	$1,096933$
	$0,132301$	$0,144147$
	$0,708820$	$0,758920$
3	$-0,650779$	$0,795222$
	$-0,156370$	$0,163747$
	$0,373489$	$0,676938$
	$0,7977296$	$-0,364092$

Tabulka 3.3: Tabulka Gaussovy-Jacobiho kvadraturní formule pro hodnotu parametrů $\alpha = 1$, $\beta = 2$ stupně n

Gaussova-Legendrova kv. f.		
n	x_i	A_i
1	$\pm\frac{\sqrt{3}}{3} \doteq 0,577350$	1
2	0	$\frac{8}{9}$
	$\pm\sqrt{\frac{3}{5}}$	$\frac{5}{9}$
3	$\pm\sqrt{\frac{3-2\sqrt{\frac{6}{5}}}{7}} \doteq \pm 0,339981$	$0,652145$
	$\pm\sqrt{\frac{3+2\sqrt{\frac{6}{5}}}{7}} \doteq \pm 0,861136$	$0,347855$

Tabulka 3.4: Tabulka Gaussovy-Legendrovy kvadraturní formule stupně n

Gaussova-Gegenbauerova kv. f.		
n	x_i	A_i
1	$\pm\frac{\sqrt{5}}{5} \pm 0,447214$	1
2	0	$\frac{4}{9}$
	$\pm 0,654654$	$\frac{7}{9}$
3	$\pm 0,285232$	$\frac{1}{2}$
	$\pm 0,765055$	$\frac{1}{2}$

Tabulka 3.5: Tabulka Gaussovy-Gegenbauerovy kvadraturní formule pro hodnotu parametru $\alpha = 1$ stupně n

Gaussova-Čebyševova kv.f. 1. typu		
n	x_i	A_i
1	$\pm\frac{\sqrt{2}}{2} \doteq \pm 0,707107$	$\frac{\pi}{2}$
2	0	$\frac{\pi}{3}$
	$\pm\frac{\sqrt{3}}{2} \doteq \pm 0,866025$	
3	$\pm\frac{\sqrt{2+\sqrt{2}}}{2} \doteq \pm 0,92386$	$\frac{\pi}{4}$
	$\pm\frac{\sqrt{2-\sqrt{2}}}{2} \doteq \pm 0,38268$	

Tabulka 3.6: Tabulka Gaussovy-Čebyševovy kvadraturní formule 1. typu stupně n

Gaussova-Čebyševova kv. f. 2. typu		
n	x_i	A_i
1	$\pm\frac{1}{2} = 0,5$	1
2	0 $\pm\frac{\sqrt{2}}{2} \doteq \pm 0,707107$	$\frac{2}{3}$
3	$\pm\frac{1-\sqrt{5}}{4} \doteq \mp 0,309017$ $\pm\frac{1+\sqrt{5}}{4} \doteq \pm 0,809017$	$\pm 0,937897$ $\pm 0,062103$

Tabulka 3.7: Tabulka Gaussovy-Čebyševovy kvadraturní formule 2. typu stupně n

Poznámka. Informace z této kapitoly a také mnohé další týkající se numerického integrování lze najít v publikacích [2], [6].

3.3 Symetrické diferenciální operátory

Jednou ze zajímavostí je samotný původ většiny ortogonálních polynomů. Ukážeme si vznik ortogonálních polynomů jakožto systému vlastních funkcí nějakého symetrického diferenciálního operátoru.

Podrobnější informace k tomuto tématu lze najít v [7].

Definice 3.8. Nechť V je reálný vektorový prostor se skalárním součinem a $L : V \to V$ je lineární zobrazení, přičemž definiční obor L nemusí být celý prostor V. Nenulová funkce $f \in V$ je vlastní vektor příslušný číslu λ, jestliže

$$Lf = \lambda f.$$

L je symetrický operátor, jestliže platí

$$\langle Lf, g\rangle = \langle f, Lg\rangle$$

pro $\forall f, g \in V$, pro něž je L definován.

Věta 3.12. *Nechť L je symetrický lineární operátor definovaný na vektorovém prostoru V se skalárním součinem. Jsou-li f_1, f_2 vlastní vektory odpovídající různým vlastním číslům λ_1 ,λ_2, pak jsou vektory f_1, f_2 ortogonální.*

Důkaz. Nechť $Lf_1 = \lambda_1 f_1$, $Lf_2 = \lambda_1 f_2$. Pak platí

$$\langle Lf_1, f_2\rangle = \langle \lambda_1 f_1, f_2\rangle = \lambda_1 \langle f_1, f_2\rangle$$

$$\langle f_1, Lf_2\rangle = \langle f_1, \lambda_2 f_2\rangle = \lambda_2 \langle f_1, f_2\rangle$$

Jelikož ale $\lambda_1 \neq \lambda_2$, musí platit, že $\langle f_1, f_2\rangle = 0$. □

Významnou skupinu ortogonálních polynomů tvoří ty, které jsou řešením speciálního typu diferenciální rovnice. Ta je tvaru

$$Q(x)f'' + L(x)f' + \mu f = 0,$$

kde $Q(x)$ je polynom (nejčastěji kvadratický) a $L(x)$ je lineární polynom. Funkce f a konstanta μ jsou neznámé. Jedná se o Sturmův-Liouvillův typ diferenciální rovnice. Řešení rovnice se dá tedy přeformulovat na hledání vlastního vektoru f splňujícího

$$D(f) = Q(x)f'' + L(x)f',$$

který dostaneme odečtením členu μf. Úkolem nyní je nalézt vlastní vektory (neboli vlastní funkce) f a příslušná vlastní čísla $\lambda = -\mu$ tak, aby vlastní funkce neměla body nespojitosti a zároveň platilo, že

$$D(f) = \lambda f.$$

Diferenciální operátor $D(f)$ se nazývá Sturmův-Liouvillův operátor.

Zamezení výskytu singularit lze zajistit splněním některé z těchto podmínek:

(a) $Q(x)$ je kvadratický polynom, $L(x)$ je lineární, $Q(x)$ má dva jednoduché reálné kořeny, mezi kterými leží kořen $L(x)$ a vedoucí koeficienty polynomů $Q(x)$ a $L(x)$ mají stejné znaménko.

(b) $Q(x)$ a $L(x)$ jsou oba lineární polynomy, jejich kořeny jsou navzájem různé a vedoucí koeficienty polynomů $Q(x)$ a $L(x)$ mají stejné znaménko, pokud je kořen polynomu $L(x)$ menší než kořen polynomu $Q(x)$, nebo mají opačné znaménko, pokud je kořen polynom $L(x)$ větší než kořen polynomu $Q(x)$.

(c) $Q(x)$ je konstantní polynom, $L(x)$ je lineární a vedoucí koeficient polynomu $L(x)$ má opačné znaménko než $Q(x)$.

Podle toho, kterou podmínku diferenciální rovnice, potažmo diferenciální operátor, splňuje, rozlišujeme řešení na 3 typy polynomů: Jacobiho typ, Laguerrův typ a Hermitův typ.

Druhy polynomů	$Q(x)$	$L(x)$	λ_n
Jacobiho	$1-x^2$	$\beta-\alpha-(\alpha+\beta+2)x$	$-n(n+\alpha+\beta+1)$
Legendreovy	$1-x^2$	$-2x$	$-n(n+1)$
Gegenbauerovy	$1-x^2$	$-(2\alpha+1)x$	$-n(n+2\alpha)$
Čebyševovy 1.typu	$1-x^2$	$-x$	$-n^2$
Čebyševovy 2.typu	$1-x^2$	$-3x$	$-n(n+2)$
Laquerrovy	x	$\alpha-x+1$	$-n$
Hermitovy	1	$-2x$	$-2n$

Tabulka 3.8: Tabulka funkcí $Q(x)$, $L(x)$ pro jednotlivé typy polynomů

Jednotlivé vztahy pro λ_n si ověříme výpočtem. Do příslušné diferenciální rovnice dosadíme polynom stupně n a porovnáním koeficientů u nejvyšší mocniny získáme podmínku pro vlastní čísla diferenciálního operátoru. Vzhledem k linearitě operátoru $D(y)$ stačí ověřit pro polynom $y = x^n$.

Jacobiho polynomy:

$$D(y) = (1-x^2)y'' + [\beta - \alpha - (\alpha + \beta + 2)x]y'.$$

Dosadíme za $y = x^n$:

$$\begin{aligned} D(y) &= (1-x^2)\cdot[x^n]'' + [\beta - \alpha - (\alpha+\beta+2)x]\cdot[x^n]' \\ &= (1-x^2)n(n-1)x^{n-2} + [\beta-\alpha-(\alpha+\beta+2)x]nx^{n-1} \\ &= -[n(n+\alpha+\beta+1)]x^n + n(\beta-\alpha)x^{n-1} + n(n-1)x^{n-2}. \end{aligned}$$

Jestliže koeficient u nejvyšší mocniny polynomu stupně n p_n je roven a_n, pak $D(p_n)$ je také polynom stupně n a má koeficient u nejvyšší mocniny roven $\lambda_n = -n(n+\alpha+\beta+1)a_n$.

Legendreovy polynomy:

$$D(y) = (1-x^2)y'' - 2xy'.$$

Dosadíme za $y = x^n$:

$$\begin{aligned} D(y) &= (1-x^2)\cdot[x^n]'' - 2x\cdot[x^n]' = (1-x^2)n(n-1)x^{n-2} - 2nx^n \\ &= -n(n+1)x^n + n(n-1)x^{n-2}. \end{aligned}$$

Jestliže koeficient u nejvyšší mocniny polynomu stupně n p_n je roven a_n, pak $D(p_n)$ je také polynom stupně n a má koeficient u nejvyšší mocniny roven $-n(n+1)a_n$.

Gegenbauerovy polynomy:

$$D(y) = (1-x^2)y'' - (2\alpha+1)xy'.$$

Dosadíme za $y = x^n$:

$$\begin{aligned} D(y) &= (1-x^2)\cdot[x^n]'' - (2\alpha+1)x\cdot[x^n]' = (1-x^2)n(n-1)x^{n-2} - (2\alpha+1)nx^n \\ &= -n(n+2\alpha)x^n + n(n-1)x^{n-2}. \end{aligned}$$

Jestliže koeficient u nejvyšší mocniny polynomu stupně n p_n je roven a_n, pak $D(p_n)$ je také polynom stupně n a má koeficient u nejvyšší mocniny roven $-n(n+2\alpha)a_n$.

Čebyševovy polynomy 1.typu:

$$D(y) = (1-x^2)y'' - xy'.$$

Dosadíme za $y = x^n$:

$$\begin{aligned} D(y) &= (1-x^2)\cdot[x^n]'' - x\cdot[x^n]' = (1-x^2)n(n-1)x^{n-2} - nx^n \\ &= -n^2x^n + n(n-1)n^{n-2}. \end{aligned}$$

Jestliže koeficient u nejvyšší mocniny polynomu stupně n p_n je roven a_n, pak $D(p_n)$ je také polynom stupně n a má koeficient u nejvyšší mocniny roven $-n^2a_n$.

Čebyševovy polynomy 2.typu:

$$D(y) = (1-x^2)y'' - 3xy'.$$

Dosadíme za $y = x^n$:

$$\begin{aligned} D(y) &= (1-x^2)\cdot[x^n]'' - 3x\cdot[x^n]' = (1-x^2)n(n-1)x^n - 3nx^n \\ &= -n(n+2)x^n + n(n-1)x^{n-2}. \end{aligned}$$

Jestliže koeficient u nejvyšší mocniny polynomu stupně n p_n je roven a_n, pak $D(p_n)$ je také polynom stupně n a má koeficient u nejvyšší mocniny roven $-n(n+2)a_n$.

Laquerrovy polynomy:

$$D(y) = xy'' + (\alpha - x + 1)y'.$$

Dosadíme za $y = x^n$:

$$\begin{aligned} D(y) &= x\cdot[x^n]'' + (\alpha - x + 1)\cdot[x^n]' = xn(n-1)x^{n-2} + (\alpha - x + 1)nx^{n-1} \\ &= -nx^n + (\alpha + n)x^{n-1}. \end{aligned}$$

Jestliže koeficient u nejvyšší mocniny polynomu stupně n p_n je roven a_n, pak $D(p_n)$ je také polynom stupně n a má koeficient u nejvyšší mocniny roven $-na_n$.

Hermitovy polynomy:

$$D(y) = y'' - 2xy'.$$

Dosadíme za $y = x^n$:

$$D(y) = [x^n]'' - 2x\cdot[x^n]' = n(n-1)x^{n-2} - 2x\cdot nx^{n-1} = -2nx^n + n(n-1)x^{n-2}.$$

Jestliže koeficient u nejvyšší mocniny polynomu stupně n p_n je roven a_n, pak $D(p_n)$ je také polynom stupně n a má koeficient u nejvyšší mocniny roven $-2na_n$.

Pro všechny tyto typy platí následující vlastnosti:

- Řešením je posloupnost polynomů $P_0, P_1, P_2, \ldots$, přičemž každý polynom P_n je stupně právě n a odpovídá vlastnímu číslu λ_n.
- Interval $[a,b]$, na kterém jsou všechny vlastní funkce ortogonální, je ohraničen kořeny polynomu $Q(x)$.
- Kořen polynomu $L(x)$ se nachází uvnitř intervalu$[a,b]$.
- Vahová funkce nenabývá nulových či nevlastních hodnot uvnitř intervalu $[a,b]$.
- Diferenciální rovnici lze upravit tak, aby vahová funkce byla uvnitř intervalu $[a,b]$ kladná.
- Pokud $R(x) = e^{\int \frac{L(x)}{Q(x)}\,\mathrm{d}x}$, pak jsou polynomy odpovídající dané diferenciální rovnici ortogonální s vahou $w(x) = \frac{R(x)}{Q(x)}$.

Závěr

V bakalářské práci byly shrnuty základní vlastnosti ortogonálních polynomů a systémů v rozsahu nutném pro jejich použití. Bylo představeno rozdělení nejznámějších typů a pro každý typ byly uvedeny jeho vlastnosti a vykreslen graf prvních několika polynomů. Na závěr byly vysvětleny tři možnosti použití ortogonálních polynomů, a to aproximace funkcí, numerické integrování a použití jako symetrického diferenciálního operátoru.
V každé části byly uvedeny klíčové znalosti a v prvních dvou částech také uvedeny příklady výpočtu.

Pro každý zmiňovaný typ polynomu pak byl vytvořen zdrojový kód softwaru MATLAB pro výpočet polynomu zvoleného stupně, případně i parametrů. Dále byly vytvořeny programy pro vykreslení grafu vypočteného polynomu.

Do budoucna by bylo vhodné vytvořit širší databázi druhů ortogonálních polynomů, která by zároveň mohla obsahovat i další vlastnosti, které v práci nejsou zmíněny.

Také by mohl být vytvořen softwarový balík pro samotnou aplikaci ortogonálních polynomů, např. pro aproximaci funkce či numerický výpočet integrálu.

Příloha

Zde se nachází kompletní zdrojový kód vytvořených programů pro výpočet polynomů a tvorbu grafů v programu MATLAB.

Hermitovy polynomy

```
function[Hn,HnS]=pHer(n)
%Funkce pHer pocita Hermitovy polynomy n-teho radu.
x=sym('x'); %promenna x symbolickeho typu
if n==0 %cyklus pro vypocet polynomu pomoci rekurentniho vzorce
    HnS=1; %symbolicky vystup
    Hn=1; %vystup ve tvaru polynomu v MATLABu
elseif n==1
    HnS=2*x; %symbolicky vystup
    Hn=sym2poly(HnS); %vystup ve tvaru polynomu v MATLABu
elseif n>=2 && (round(n)-n==0)
    Hi=1;
    Hii=2*x;
    for i=1:(n-1)
    Hiii=2*x*Hii-2*i*Hi;
    Hi=Hii;
    Hii=Hiii;
   end;
   HnS=simplify(Hiii); %symbolicky vystup
   Hn=sym2poly(Hiii); %vystup ve tvaru polynomu v MATLABu
else error('Zadany rad neni celociselny ci nezaporny !!!')
end;

function[Hn,HnS]=pHerG(n)
%Funkce pHerG pocita Hermitovy polynomy n-teho radu a vykresluje graf
daneho polynomu.
x=sym('x'); %promenna x symbolickeho typu
if n==0 %cyklus pro vypocet polynomu pomoci rekurentniho vzorce
    HnS=1; %symbolicky vystup
    Hn=1; %vystup ve tvaru polynomu v MATLABu
    plot(linspace(-1,1,100),polyval(Hn,linspace(-1,1,100))); %graf
elseif n==1
    HnS=2*x; %symbolicky vystup
```

```
    Hn=sym2poly(HnS);  %vystup ve tvaru polynomu v MATLABu
    plot(linspace(-1,1,100),polyval(Hn,linspace(-1,1,100))); %graf
elseif n>=2 && (round(n)-n==0)
    Hi=1;
    Hii=2*x;
    for i=1:(n-1)
    Hiii=2*x*Hii-2*i*Hi;
    Hi=Hii;
    Hii=Hiii;
   end;
   HnS=simplify(Hiii); %symbolicky vystup
   Hn=sym2poly(Hiii); %vystup ve tvaru polynomu v MATLABu
   a=linspace(-ceil(max(roots(Hn))),ceil(max(roots(Hn))),100);
%vypocet intervalu
   plot(a,polyval(Hn,a)); %graf
else error('Zadany rad neni celociselny ci nezaporny !!!')
end;
end
```

Laguerrovy polynomy

```
function[Lna,LnaS]=pLag(n,alpha)
%Funkce pLag pocita Laguerrovy polynomy n-teho radu.
x=sym('x'); %promenna x symbolickeho typu
if n==0  && alpha>-1 %cyklus pro vypocet polynomu pomoci
 rekurentniho vzorce
    LnaS=1; %symbolicky vystup
    Lna=1; %vystup ve tvaru polynomu v MATLABu
elseif n==1 && alpha>-1
    LnaS=-x+1+alpha; %symbolicky vystup
    Lna=sym2poly(LnaS); %vystup ve tvaru polynomu v MATLABu
elseif n>=2 && (round(n)-n==0) && alpha>-1
    Li=1;
    Lii=-x+alpha+1;
    for i=1:(n-1)
    Liii=(2*i+alpha+1-x)/(i+1)*Lii-((i+alpha)/(i+1))*Li;
    Li=Lii;
    Lii=Liii;
   end;
   LnaS=simplify(Liii); %symbolicky vystup
   Lna=sym2poly(Liii); %vystup ve tvaru polynomu v MATLABu
else
    if alpha>-1 %test na zadane parametry
        error('Zadany rad neni celociselny ci nezaporny !!!');
    elseif (round(n)-n==0) && n>=0
        error('Nevhodne zvoleny koeficient!!!');
```

```
    else
        error('Zadany rad neni celociselny ci nezaporny a nevhodne
 zvoleny koeficienty!!!')
    end;
end;

function[Lna,LnaS]=pLagG(n,alpha)
%Funkce pLeg pocita Laguerrovy polynomy n-teho radu a vykresluje graf
daneho polynomu.
x=sym('x'); %promenna x symbolickeho typu
if n==0  && alpha>-1 %cyklus pro vypocet polynomu pomoci
 rekurentniho vzorce
    LnaS=1; %symbolicky vystup
    Lna=1; %vystup ve tvaru polynomu v MATLABu
    plot(linspace(0,1,100),polyval(Lna,linspace(0,1,100))); %graf
elseif n==1 && alpha>-1
    LnaS=-x+1+alpha; %symbolicky vystup
    Lna=sym2poly(LnaS); %vystup ve tvaru polynomu v MATLABu
    plot(linspace(0,1,100),polyval(Lna,linspace(0,1,100))); %graf
elseif n>=2 && (round(n)-n==0) && alpha>-1
    Li=1;
    Lii=-x+alpha+1;
    for i=1:(n-1)
    Liii=(2*i+alpha+1-x)/(i+1)*Lii-((i+alpha)/(i+1))*Li;
    Li=Lii;
    Lii=Liii;
   end;
   LnaS=simplify(Liii); %symbolicky vystup
   Lna=sym2poly(Liii); %vystup ve tvaru polynomu v MATLABu
   a=linspace(0,ceil(max(roots(Lna))),100); %interval pro graf
   plot(a,polyval(Lna,a)); %vykresleni grafu
else
    if alpha>-1 %test na zadane parametry
        error('Zadany rad neni celociselny ci nezaporny !!!');
    elseif (round(n)-n==0) && n>=0
        error('Nevhodne zvoleny koeficient!!!');
    else
        error('Zadany rad neni celociselny ci nezaporny a nevhodne
 zvoleny koeficienty!!!')
    end;
end;
```

Jacobiho polynomy

```
function[Pn,PnS]=pJac(n,a,b)
%Funkce pJac pocita Jacobiho polynomy n-teho radu. Koeficinety a, b
```

```
 jsou soucasti rekurentniho vztahu pro vypocet polynomu.
x=sym('x'); %promenna x symbolickeho typu
if n==0 && a>-1 && b>-1 %cyklus pro vypocet polynomu pomoci
 rekurentniho vzorce
    Pn=1; %vystup ve tvaru MATLABu
    PnS=1; %symbolicky vystup
elseif n==1 && a>-1 && b>-1
    PnS=0.5*(2*(a+1)+(a+b+2)*(x-1)); %symbolicky vystup
    Pn=sym2poly(PnS); %vystup ve tvaru MATLABu
elseif n>=2 && (round(n)-n==0) && a>-1 && b>-1
    Pi=1;
    Pii=0.5*(2*(a+1)+(a+b+2)*(x-1));
    for i=1:(n-1)
    Piii=((2*i+a+b+1)*((2*i+a+b+2)*(2*i+a+b)*x+a^2-b^2)*Pii)/(2*(i+1)
*(i+a+b+1)*(2*i+a+b))-((2*(i+a)*(i+b)*(2*i+a+b+2)*Pi)/(2*(i+1)
*(i+a+b+1)*(2*i+a+b)));
    Pi=Pii;
    Pii=Piii;
   end;
   PnS=simplify(Piii); %symbolicky vystup
   Pn=sym2poly(Piii); %vystup ve tvaru MATLABu
else
    if a>-1 && b>-1 %test zadanych parametru
        error('Zadany rad neni celociselny ci nezaporny !!!');
    elseif (round(n)-n==0) && n>=0
        error('Nevhodne zvolene koeficienty!!!');
    else
        error('Zadany rad neni celociselny ci nezaporny a nevhodne
 zvolene koeficienty!!!')
    end;
end;

function[Pn,PnS]=pJacG(n,a,b)
%Funkce pJacG pocita Jacobiho polynomy n-teho radu a vykresluje graf
daneho polynomu. Koeficinety a, b jsou soucasti rekurentniho vztahu
 pro vypocet polynomu.
x=sym('x'); %promenna x symbolickeho typu
d=linspace(-1,1,100); %interval pro graf
if n==0 && a>-1 && b>-1 %cyklus pro vypocet polynomu pomoci
 rekurentniho vzorce
    Pn=1; %vystup ve tvaru MATLABu
    PnS=1; %symbolicky vystup
elseif n==1 && a>-1 && b>-1
    PnS=0.5*(a+b+2)*x+0.5*(a-b); %symbolicky vystup
    Pn=sym2poly(PnS); %vystup ve tvaru MATLABu
elseif n>=2 && (round(n)-n==0) && a>-1 && b>-1
```

```
    Pi=1;
    Pii=0.5*(2*(a+1)+(a+b+2)*(x-1));
    for i=1:(n-1)
    Piii=((2*i+a+b+1)*((2*i+a+b+2)*(2*i+a+b)*x+a^2-b^2)*Pii)
/(2*(i+1)*(i+a+b+1)*(2*i+a+b))-((2*(i+a)*(i+b)*(2*i+a+b+2)*Pi)
/(2*(i+1)*(i+a+b+1)*(2*i+a+b)));
    Pi=Pii;
    Pii=Piii;
   end;
   PnS=simplify(Piii); %symbolicky vystup
   Pn=sym2poly(Piii); %vystup ve tvaru polynomu v MATLABu
else
    if a>-1 && b>-1 %test zadanych parametru
        error('Zadany rad neni celociselny ci nezaporny !!!');
    elseif (round(n)-n==0) && n>=0
        error('Nevhodne zvolene koeficienty!!!');
    else
        error('Zadany rad neni celociselny ci nezaporny a nevhodne
 zvolene koeficienty!!!')
    end;
end;
plot(d,polyval(Pn,d)); %vykresleni grafu
end
```

Legendrovy polynomy

```
function[Pn,PnS]=pLeg(n)
%Funkce pLeg pocita Legendrovy polynomy n-teho radu.
x=sym('x'); %promenna x symbolickeho typu
if n==0 %cyklus pro vypocet polynomu pomoci rekurentniho vzorce
        PnS=1; %symbolicky vystup
        Pn=1; %vystup ve tvaru polynomu v MATLABu
elseif n==1
        PnS=x; %symbolicky vystup
        Pn=sym2poly(PnS); %vystup ve tvaru polynomu v MATLABu
elseif n>=2 && (round(n)-n==0)
        Pi=1;
        Pii=x;
        for i=1:(n-1)
            Piii=(2*i+1)/(i+1)*x*Pii-(i/(i+1))*Pi;
            Pi=Pii;
            Pii=Piii;
        end;
        PnS=simplify(Piii); %symbolicky vystup
        Pn=sym2poly(Piii); %vystup ve tvaru polynomu v MATLABu
else error('Zadany rad neni celociselny ci nezaporny !!!');
```

```
end;

function[Pn,PnS]=pLegG(n)
%Funkce pLegG pocita Legendrovy polynomy n-teho radu a vykresluje
graf daneho polynomu.
x=sym('x'); %promenna x symbolickeho typu
if n==0 %cyklus pro vypocet polynomu pomoci rekurentniho vzorce
    PnS=1; %symbolicky vystup
    Pn=1; %vystup ve tvaru polynomu v MATLABu
    plot(linspace(-1,1,100),polyval(Pn,linspace(-1,1,100))); %graf
elseif n==1
    PnS=x; %symbolicky vystup
    Pn=sym2poly(PnS); %vystup ve tvaru polynomu v MATLABu
    plot(linspace(-1,1,100),polyval(Pn,linspace(-1,1,100))); %graf
elseif n>=2 && (round(n)-n==0)
    Pi=1;
    Pii=x;
    for i=1:(n-1)
    Piii=(2*i+1)/(i+1)*x*Pii-(i/(i+1))*Pi;
    Pi=Pii;
    Pii=Piii;
   end;
   PnS=simplify(Piii); %symbolicky vystup
   Pn=sym2poly(Piii); %vystup ve tvaru polynomu v MATLABu
   plot(linspace(-1,1,100),polyval(Pn,linspace(-1,1,100))); %graf
else error('Zadany rad neni celociselny ci nezaporny !!!')
end;
```

Gegenbauerovy polynomy

```
function[Cn,CnS]=pGeg(n,alpha)
%Funkce pGeg pocita Gegenbauerovy polynomy n-teho radu. Koeficient
 alpha je soucasti rekurentniho vztahu pro vypocet polynomu.
x=sym('x'); %promenna x symbolickeho typu
if n==0 && alpha>-0.5 %cyklus pro vypocet polynomu pomoci
 rekurentniho vzorce
    Cn=1; %vystup ve tvaru polynomu v MATLABu
    CnS=1; %symbolicky vystup
elseif n==1 && alpha>-0.5
    CnS=2*alpha*x; %symbolicky vystup
    Cn=sym2poly(CnS); %vystup ve tvaru polynomu v MATLABu
elseif n>=2 && (round(n)-n==0) && alpha>-0.5
    Ci=1;
    Cii=2*alpha*x;
    for i=1:(n-1)
    Ciii=(2*x*(i+alpha)*Cii-(i+2*alpha-1)*Ci)/(i+1);
```

```
    Ci=Cii;
    Cii=Ciii;
   end;
   CnS=simplify(Ciii); %symbolicky vystup
   Cn=sym2poly(Ciii); %vystup ve tvaru polynomu v MATLABu
else
    if alpha>-0.5 %test na zadane parametry
        error('Zadany rad neni celociselny ci nezaporny !!!');
    elseif (round(n)-n==0) && n>=0
        error('Nevhodne zvoleny koeficient!!!');
    else
        error('Zadany rad neni celociselny ci nezaporny a nevhodne
 zvoleny koeficienty!!!')
    end;
end;

function[Cn,CnS]=pGegG(n,alpha)
%Funkce pGegG pocita Gegenbauerovy polynomy n-teho radu a vykresluje
 graf daneho polynomu. Koeficient alpha je soucasti rekurentniho
vztahu pro vypocet polynomu.
x=sym('x'); %promenna x symbolickeho typu
a=linspace(-1,1,100); %interval pro graf
if n==0 && alpha>-0.5 %cyklus pro vypocet polynomu pomoci
 rekurentniho vzorce
    Cn=1; %vystup ve tvaru polynomu v MATLABu
    CnS=1; %symbolicky vystup
elseif n==1 && alpha>-0.5
    CnS=2*alpha*x; %symbolicky vystup
    Cn=sym2poly(CnS); %vystup ve tvaru polynomu v MATLABu
elseif n>=2 && (round(n)-n==0) && alpha>-0.5
    Ci=1;
    Cii=2*alpha*x;
    for i=1:(n-1)
    Ciii=(2*x*(i+alpha)*Cii-(i+2*alpha-1)*Ci)/(i+1);
    Ci=Cii;
    Cii=Ciii;
   end;
   CnS=simplify(Ciii); %symbolicky vystup
   Cn=sym2poly(Ciii); %vystup ve tvaru polynomu v MATLABu
else
    if alpha>-0.5 %test na zadane parametry
        error('Zadany rad neni celociselny ci nezaporny !!!');
    elseif (round(n)-n==0) && n>=0
        error('Nevhodne zvoleny koeficient!!!');
    else
        error('Zadany rad neni celociselny ci nezaporny a nevhodne
```

```
 zvoleny koeficienty!!!')
    end;
end;
plot(a,polyval(Cn,a)); %vykresleni grafu
end
```

Čebyševovy polynomy 1. typu

```
function[Tn,TnS]=pCeb(n)
%Funkce pCeb pocita Cebysevovy polynomy n-teho radu.
x=sym('x'); %promenna x symbolickeho typu
if n==0 %cyklus pro vypocet polynomu pomoci rekurentniho vzorce
    Tn=1; %symbolicky vystup
    TnS=1; %vystup ve tvaru polynomu v MATLABu
elseif n==1
    TnS=x; %symbolicky vystup
    Tn=sym2poly(TnS); %vystup ve tvaru polynomu v MATLABu
elseif n>=2 && (round(n)-n==0)
    Ti=1;
    Tii=x;
    for i=1:(n-1)
    Tiii=2*x*Tii-Ti;
    Ti=Tii;
    Tii=Tiii;
   end;
   TnS=simplify(Tiii); %symbolicky vystup
   Tn=sym2poly(Tiii); %vystup ve tvaru polynomu v MATLABu
else error('Zadany rad neni celociselny ci nezaporny !!!')
end;

function[Tn,TnS]=pCebG(n)
%Funkce pCebG pocita Cebysevovy polynomy n-teho radu a vykresluje
 graf daneho polynomu.
x=sym('x'); %promenna x symbolickeho typu
a=linspace(-1,1,100); %interval pro graf
if n==0 %cyklus pro vypocet polynomu pomoci rekurentniho vzorce
    TnS=1; %symbolicky vystup
    Tn=1; %vystup ve tvaru polynomu v MATLABu
elseif n==1
    TnS=x; %symbolicky vystup
    Tn=sym2poly(TnS); %vystup ve tvaru polynomu v MATLABu
elseif n>=2 && (round(n)-n==0)
    Ti=1;
    Tii=x;
    for i=1:(n-1)
    Tiii=2*x*Tii-Ti;
```

```
    Ti=Tii;
    Tii=Tiii;
    end;
   TnS=simplify(Tiii); %symbolicky vystup
   Tn=sym2poly(Tiii); %vystup ve tvaru polynomu v MATLABu
else error('Zadany rad neni celociselny ci nezaporny !!!')
end;
plot(a,polyval(Tn,a)); %vykresleni grafu
end
```

Čebyševovy polynomy 2. typu

```
function[Un,UnS]=pCeb2(n)
%Funkce pCeb pocita Cebysevovy polynomy druheho typu n-teho radu.
x=sym('x'); %promenna x symbolickeho typu
if n==0 %cyklus pro vypocet polynomu pomoci rekurentniho vzorce
    Un=1; %symbolicky vystup
    UnS=1; %vystup ve tvaru polynomu v MATLABu
elseif n==1
    UnS=2*x; %symbolicky vystup
    Un=sym2poly(UnS); %vystup ve tvaru polynomu v MATLABu
elseif n>=2 && (round(n)-n==0)
    Ui=1;
    Uii=2*x;
    for i=1:(n-1)
    Uiii=2*x*Uii-Ui;
    Ui=Uii;
    Uii=Uiii;
   end;
   UnS=simplify(Uiii); %symbolicky vystup
   Un=sym2poly(Uiii); %vystup ve tvaru polynomu v MATLABu
else error('Zadany rad neni celociselny ci nezaporny !!!')
end;

function[Un,UnS]=pCeb2G(n)
%Funkce pCeb2G pocita Cebysevovy polynomy druheho typu n-teho radu a
 vykresluje graf daneho polynomu.
x=sym('x'); %promenna x symbolickeho typu
a=linspace(-1,1,10000); %interval pro graf
if n==0 %cyklus pro vypocet polynomu pomoci rekurentniho vzorce
    Un=1; %vystup ve tvaru polynomu v MATLABu
    UnS=1; %symbolicky vystup
elseif n==1
    UnS=2*x; %symbolicky vystup
    Un=sym2poly(UnS); %vystup ve tvaru polynomu v MATLABu
elseif n>=2 && (round(n)-n==0)
```

```
    Ui=1;
    Uii=2*x;
    for i=1:(n-1)
    Uiii=2*x*Uii-Ui;
    Ui=Uii;
    Uii=Uiii;
   end;
   UnS=simplify(Uiii); %symbolicky vystup
   Un=sym2poly(Uiii); %vystup ve tvaru polynomu v MATLABu
else error('Zadany rad neni celociselny ci nezaporny !!!')
end;
plot(a,polyval(Un,a)); %vykresleni grafu
end
```

Seznam použité literatury

[1] Szego, Gabor, *Orthogonal polynomials*. Vydání 4. Providence: American Mathematical Society, 1939 – 1975, xiii, 432 p. Colloquium publications (American Mathematical Society), v. 23. ISBN 0-8218-1023-5.

[2] Horová, Ivana a Jiří Zelinka. *Numerické metody*. Vyd. 2., rozš. Brno: Masarykova Univerzita v Brně, 2004, vi, 285 s. ISBN 80-210-3317-7.

[3] Abramowitz, Milton; Stegun, Irene A., eds. (1965), „Chapter 22“, *Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables*, New York: Dover, p. 773, ISBN 978-0486612720, MR 0167642.

[4] Koornwinder, Tom H.; Wong, Roderick S. C.; Koekoek, Roelof; Swarttouw, René F. (2010), *Orthogonal Polynomials*, v Olver, Frank W. J.; Lozier, Daniel M.; Boisvert, Ronald F.; Clark, Charles W., *NIST Handbook of Mathematical Functions*, Cambridge University Press, ISBN 978-0521192255, MR 2723248

[5] Kolář, Martin. *Spektrální analýza 1*. Výukový materiál Masarykovy univerzity, 41 s. Dostupné z: http://www.math.muni.cz/ mkolar/fa.pdf

[6] Kolář, Martin. *Spektrální analýza 2*. Výukový materiál Masarykovy univerzity, 44 s. Dostupné z: http://www.math.muni.cz/ mkolar/SAII.pdf

[7] *Orthogonal polynomials*. Výukový materiál Umm Al-Qura University, 22 s. Dostupné z: https://uqu.edu.sa/page/ar/157410

[8] Zelinka, Jiří a Jan Koláček. *Jak pracovat s MATLABem*. Výukový materiál, 40 s. Dostupné z: https://www.math.muni.cz/ kolacek/vyuka/vypsyst/navod.pdf

[9] Cautschi, Walter. Orthogonal Polynomials (in Matlab). Výukový materiál, 23 s. Dostupné z: https://www.cs.purdue.edu/homes/wxg/OPmatlab.pdf