

Polinomna interpolacija i njena primjena u elektrotehnici

Petrović, Luka

Undergraduate thesis / Završni rad

2023

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **University of Rijeka, Faculty of Engineering / Sveučilište u Rijeci, Tehnički fakultet**

Permanent link / Trajna poveznica: <https://um.nsk.hr/um:nbn:hr:190:349618>

Rights / Prava: [Attribution 4.0 International](#)/[Imenovanje 4.0 međunarodna](#)

Download date / Datum preuzimanja: **2024-07-17**



Repository / Repozitorij:

[Repository of the University of Rijeka, Faculty of Engineering](#)



SVEUČILIŠTE U RIJECI

TEHNIČKI FAKULTET

Prijediplomski sveučilišni studij elektrotehnike

Završni rad

**POLINOMNA INTERPOLACIJA I NJENA PRIMJENA U
ELEKTROTEHNICI**

Rijeka, srpanj 2023.

Luka Petrović
0069079539

SVEUČILIŠTE U RIJECI

TEHNIČKI FAKULTET

Prijediplomski sveučilišni studij elektrotehnike

Završni rad

**POLINOMNA INTERPOLACIJA I NJENA PRIMJENA U
ELEKTROTEHNICI**

Mentor: izv. prof. dr. sc. Ivan Dražić

Komentor: prof. dr. sc. Viktor Sučić

Rijeka, srpanj 2023.

Luka Petrović
0069079539

Na mjesto ove stranice umetnuti potpisani izvornik zadatka!

IZJAVA

Sukladno članku 7. stavku 1. Pravilnika o završnom radu, završnom ispitu i završetku sveučilišnih prijediplomskih studija Tehničkog fakulteta Sveučilišta u Rijeci od 4. travnja 2023., izjavljujem da sam samostalno izradio završni rad prema zadatku preuzetom dana 15. ožujka 2021.

Rijeka, 12.07.2023.

Luka Petrović

Posvećujem ovaj završni rad svojim roditeljima, braći, i djevojci Ivani. Veliko hvala svima na bezuvjetnoj podršci i potpori tijekom mog temeljitog studiranja. Posebno hvala mentoru izv. prof. dr. sc. Ivanu Dražiću na savjetima, razumijevanju i suradnji pri izradi ovog završnog rada.

Sadržaj

1. Uvod	3
2. Problem aproksimacije	4
2.1. Greške	4
2.2. Apsolutna i relativna greška	5
2.3. Greške metode	6
2.3.1. Greške diskretizacije	6
2.3.2. Greške odbacivanja	7
2.3.3. Funkcija greške	7
3. Interpolacija	8
3.1. Definiranje interpolacije	8
3.2. Normiranje aproksimacija	8
3.3. Oblici interpolacijskog polinoma	9
4. Lagrangeov oblik interpolacijskog polinoma	11
5. Newtonov oblik interpolacijskog polinoma	18
6. Spline interpolacija	27
6.1. Linerani spline	27
6.2. Kubični spline	28
7. Hermiteova interpolacija	33
7.1. Greška Hermiteove interpolacije	34
7.2. Newtonova forma Hermiteove interpolacije	35
8. Gaussove formule	38
9. Zaključak	40
Bibliografija	42
Sažetak i ključne riječi	43

Summary and key words 44

Dodatak A Lista Octave kodova korištenih za primjere 45

1. Uvod

Gledano kroz kontekst primjene u elektrotehnici, polinomna interpolacija jedna je od najvažnijih matematičkih tehnika. Kroz nekoliko različitih metoda, polinomna interpolacija omogućuje aproksimiranje nepoznatih funkcija na temelju ograničene količine poznatih podataka. Elektrotehničke discipline su brojne i raznolike, ali u suštini sve iziskuju točnost i preciznost pri podešavanju, prikupljanju, i analizi podataka i mjernih veličina, gdje metode polinomne interpolacije igraju veliku ulogu. Od mjerenja fizikalnih veličina električnih strojeva i konstruiranja karakteristika koje opisuju međusobne odnose spomenutih, do rekonstrukcije otipkanih analognih signala u telekomunikacijskim mrežama.

Cilj ovog završnog rada je ustanoviti ključne koncepte i pojmove vezane za interpolaciju, te proučiti problematiku koja se javlja pri istoj. Biti će proučeni problemi aproksimacije, te utvrđeni tipovi grešaka koje se mogu pojaviti pri aproksimaciji. Također, ustanoviti će se normirane aproksimacije. Kroz odvojena poglavlja ovog rada biti će analizirana svojstva različitih oblika interpolacijskih polinoma, izvedeni njihovi osnovni oblici, te prikazani načini kako izbjeći neželjene pojave numeričke nestabilnosti i nepoželjnih oscilacija.

Elektrotehnika je područje inženjerskih znanosti koje se uvelike oslanja na matematičke modele i metode interpolacije kako bi analizirala i sintetizirala električne mreže i sustave. Kroz praktične primjere biti će prikazana primjena pojedinih oblika interpolacijskih polinoma u elektrotehnici, te preispitana njihova praktičnost i mogućnost računalne implementacije za rješavanje stvarnih problema s kojima se susrećemo u struci.

2. Problem aproksimacije

Interpolacija, koja dolazi od latinskog *inter* (između), i *polos* (os), u suštini predstavlja svaki izračun nove točke ili podatka između dvije ili više postojećih točaka/podataka. Interpolacija ima široku primjenu u svim granama znanosti, kako prirodnim, tako i formalnim i društvenim znanostima. Poboljšava preglednost prikupljenih podataka, omogućuje i olakšava opisivanje postojećih trendova, te može pomoći u predviđanju mogućih ishoda na osnovi poznatih uvjeta (ekstrapolacija). U ovom radu definirati ćemo interpolaciju, te uz nju vezane osnovne pojmove. Također, promotriti ćemo više različitih metoda interpolacije, te ih usporediti kroz vidike praktičnosti i inženjerske iskoristivosti.

Poznate su informacije o nekoj funkciji f , definiranoj na nekom skupu $X \subseteq \mathbb{R}$. Na osnovi tih informacija funkciju f želimo zamijeniti nekom drugom funkcijom ϕ na istom skupu X , s ciljem da su f i ϕ u nekom smislu približne ili bliske. Problem se javlja u dva slučaja. U prvom slučaju poznata je funkcija f , ali njen oblik je prekomplikiran za učinkovito, odnosno smisleno računanje. Tada odabiremo ključne informacije o f i po željenim kriterijima određujemo aproksimacijsku funkciju ϕ . Pošto nam je originalna funkcija poznata, lako određujemo grešku dobivene aproksimacije. U drugom slučaju, funkcija f nije nam poznata, ali poznamo njenu vrijednost u nekoliko točaka: $x_0 < x_1 < \dots < x_n$. Aproksimacijska funkcija ϕ određuje se iz poznatih vrijednosti, koje poželjno sadržavaju i očekivani oblik ponašanja funkcije ϕ . Očito, za ovaj slučaj nije moguće odrediti grešku aproksimacije bez dodatnih informacija o originalnoj funkciji f [2]. U području poznatih podataka $[x_0, x_n]$, potrebno je funkciju f aproksimirati jednostavnijom funkcijom ϕ tako da vrijedi:

$$\phi(x_i) = f(x_i), i = 0, 1, \dots, n, \quad (2.1)$$

pri čemu su $x_i, i = 0, 1, \dots, n$ vrijednosti unutar skupa X na kojima su poznate vrijednosti funkcije f .

2.1. Greške

U praksi je često potrebno, radi pojednostavljivanja ili pak ograničenja računalne memorije, koristiti približnu vrijednost umjesto stvarne veličine. Sukladno tome, u teoriji češće promatramo idealan slučaj, oduzet od svih stvarnih nesavršenosti. Stoga je potrebno razumjeti vrste grešaka i još bitnije, kako dotične greške utječu na račun i konačan rezultat.

2.2. Apsolutna i relativna greška

Neka je x stvarna vrijednost neke poznate veličine, a x_p njena približna vrijednost, odnosno, recimo da je x_p aproksimacija x . Razliku između stvarne veličine i njene aproksimacije Δx nazivamo greška aproksimacije. Uobičajeno je danu vrijednost izraziti kao apsolutnu vrijednost, te je tada zovemo **apsolutna greška aproksimacije**

$$\Delta x = |x - x_p|. \quad (2.2)$$

U praksi se često dogodi da nam točna vrijednost x nije poznata [3], ali poznato je da se vrijednost nalazi u intervalu $[-\epsilon, \epsilon]$, za $\epsilon > 0$. Stoga možemo pisati

$$-\epsilon \leq x - x_p \leq \epsilon, \quad (2.3)$$

odnosno

$$x_p - \epsilon \leq x \leq x_p + \epsilon. \quad (2.4)$$

Dotičnu veličinu ϵ nazivamo **granica greške aproksimacije**, te je taj pojam u praksi često istoznačan s apsolutnom greškom aproksimacije Δx . $x_p - \epsilon$ najmanja, a $x_p + \epsilon$ najveća vrijednost koju aproksimacija veličine x može poprimiti, stoga u praksi često pišemo

$$x = x_p \pm \epsilon, \quad (2.5)$$

odnosno, stvarna veličina x je aproksimirana brojem x_p , te pri tome apsolutna greška aproksimacije nije veća od ϵ .

U sljedećem ćemo primjeru promotriti pojam relativne i apsolutne greške kroz praktičan problem procesa proizvodnje elektrotehničkih koaksijalnih kablova.

Primjer 2.1. *Odrađena su mjerenja širina d i duljina l nove proizvodne serije koaksijalnih kablova, te su rezultati izraženi u centimetrima i dani u tablici.*

Mjerenje	1	2	3	4
Duljina l	150,032	149,996	150,004	149,968
Širina d	2,03	1,98	1,99	2,01

Odredimo aritmetičku sredinu mjerenja duljine i širine:

$$l_a = \frac{150,032 + 149,996 + 150,004 + 149,968}{4} = 150, \quad (2.6)$$

$$d_a = \frac{2,03 + 1,97 + 1,99 + 2,01}{4} = 2. \quad (2.7)$$

Aritmetičke sredine l_a i d_a moguće su aproksimacije stvarnih duljina i širina, a uzimanjem graničnih vrijednosti iz tablice dobivamo granicu aproksimacije duljine $\epsilon_l = 0,03$ i granicu aproksimacije širine $\epsilon_d = 0,032$. U struci bi često pronašli zapis izražen kao prosječnu duljinu i dopuštenu toleranciju, npr. $l = 150 \pm 0,032$ cm i $d = 2 \pm 0,03$ cm.

U prethodnom su primjeru apsolutne granice greške duljine i širine skoro iste (0,032 i 0,03), ali ne nose istu vrijednost u smislu tolerancije zato što je širina d znatno manja od duljine l . Stoga je potrebno razumjeti **relativnu grešku**.

$$\delta x = \frac{\Delta x}{|x|}, \quad (2.8)$$

gdje je Δx apsolutna greška aproksimacije, a $|x|$ apsolutna vrijednost aproksimirane veličine. Relativna greška nije definirana za vrijednost 0, te se u struci se često izražava u postocima ili promilima [2]. U kontekstu ovog primjera, relativna greška duljine δl i relativna greška širine δd su

$$\delta l = \frac{0,032}{150} \cdot 100 = 0,0213\%,$$

$$\delta d = \frac{0,03}{2} \cdot 100 = 1,5\%.$$

Iz ovih proračuna vidimo da, makar su apsolutne greške aproksimacije skoro iste, odstupanje izraženo relativnom greškom je preko 70 puta veće.

2.3. Greške metode

Greške metode pojavljuju se kada beskonačni proces pokušamo zamjeniti, odnosno opisati konačnim. Isto tako, pojavljuju se kod računanja s veličinama definiranim limesom, poput integrala i derivacija, ili kod veličina definiranim limesima konvergentnih nizova [1]. Greške koje nastaju ovakvom supstitucijom beskonačnog s konačnim dijelimo u dvije kategorije.

2.3.1. Greške diskretizacije

Greške diskretizacije javljaju se pri diskretizaciji, odnosno zamjeni kontinuirane funkcije nekim konačnim, diskretnim skupom vrijednosti, odnosno točaka. Također, ovako definiramo greške pri zamjeni vrijednosti koja konvergira prema 0 s nekim konkretnim malim brojem (npr. $\lim_{x \rightarrow 0} [x] = 0,0001$). Isto tako definiramo greške koje nastaju pri zamjeni derivacija podijeljenom razlikom, integrala nekom kvadraturnom formulom itd. Pošto je tipičan primjer ove greške aproksimacija funkcije f na nekom intervalu $[a, b]$ vrijednostima te funkcije na određenim točkama unutar danog intervala, problematika greške diskretizacije će biti detaljnije opisana u ovom radu.

2.3.2. Greške odbacivanja

Greške odbacivanja javljaju se pri zamjeni beskonačnog niza, reda ili produkta konačnim nizom, redom ili produktom. Odnosno, greške koje se javljaju kad ograničimo beskonačni skup podataka tako da "odbacimo" ostatak koji smatramo nepotrebnim za računanje, skladištenje u memoriji itd.

2.3.3. Funkcija greške

$$e(x) = f(x) - \phi(x).$$

Funkcija greške $e(x)$ je razlika između neke funkcije $f(x)$ i njene aproksimacije $\phi(x)$. Pri tome se aproksimacijska funkcija ϕ bira po uvjetu da funkcije greške $e(x)$ što manja. Za to koristimo **normirane aproksimacije**, koje ćemo detaljnije obraditi u idućem poglavlju.

3. Interpolacija

3.1. Definiranje interpolacije

Pretpostavimo da je funkcija $f : [a, b] \rightarrow \mathbb{R}$ neprekidna i da je njena vrijednost poznata u $n + 1$ točaka (takozvani *čvorovi interpolacije*) [1]

$$a \leq x_0 < x_1 < \dots < x_n \leq b, \quad (3.1)$$

gdje su a i b granice intervala na kojem promatramo funkciju, a $[x_0, x_1, \dots, x_n]$ točke u kojima funkcija zaprima vrijednosti

$$y_i = f(x_i), i = 0, 1, \dots, n. \quad (3.2)$$

Interpolacija je proces rekonstrukcije funkcije f isključivo preko poznatih čvorova interpolacije, odnosno pronalazak nove funkcije $\phi : [a, b] \rightarrow \mathbb{R}$, tako da vrijedi $\phi(x_i) = y_i$. Pri tom se greška aproksimacije standardno izražava ili preko L_2 -norme, odnosno srednjekvadratne aproksimacije, ili L_∞ -norme, odnosno minmaks aproksimacije. [2]

3.2. Normiranje aproksimacija

Kao što je ranije spomenuto, greške aproksimacije se izražavaju preko dvije standardne norme, L_2 i L_∞ norme. Objе norme imaju zasebne oblike ovisno o tome jesu li uvjeti po kojima se funkcije f i ϕ moraju podudarati zadani diskretno ili kontinuirano.

Za kontinuirane slučajeve izrazi su

$$\|f - \phi\|_2 = \sqrt{\int_a^b (f(x) - \phi(x))^2 dx}, \quad (3.3)$$

i

$$\|f - \phi\|_\infty = \max_{x \in [a, b]} |f(x) - \phi(x)|, \quad (3.4)$$

dok su za diskretne slučajeve

$$\|f - \phi\|_2 = \sqrt{\sum_{k=1}^n (f(x_k) - \phi(x_k))^2 dx}, \quad (3.5)$$

te

$$\|f - \phi\|_\infty = \max_{k=0, \dots, n} |f(x_k) - \phi(x_k)|, \quad (3.6)$$

gdje su $f(x)$ vrijednosti stvarne zadane funkcije, a $\phi(x)$ vrijednosti rekonstruirane funkcije na rasponu $[a, b]$, odnosno na skupu točaka $k = 0, 1, \dots, n$.

3.3. Oblici interpolacijskog polinoma

Za lakše razumijevanje, postavimo prvo problem linearne interpolacije.

To je najjednostavniji primjer interpolacije, odnosno nalaženje jednadžbe pravca p koji prolazi kroz neke dvije točke (x_0, y_0) i (x_1, y_1) . Jednadžba pravca standardno se izražava kao $p(x) = a_0 + a_1x$. Preko uvjeta interpolacije, odnosno **Vandermondeove matrice**, dobivamo linearni sustav za koeficijente pravca a_0 i a_1

$$p(x_0) = y_0 = a_0 + a_1x_0, \quad (3.7)$$

$$p(x_1) = y_1 = a_0 + a_1x_1, \quad (3.8)$$

odnosno, ako izlučimo koeficijente a_0 i a_1

$$a_0 = \frac{y_0x_1 - y_1x_0}{x_1 - x_0},$$

$$a_1 = \frac{y_1 - y_0}{x_1 - x_0},$$

te na kraju, uvrstimo te koeficijente u standardnu jednadžbu pravca

$$p(x) = \frac{y_0x_1 - y_1x_0}{x_1 - x_0} + \frac{y_1 - y_0}{x_1 - x_0}x. \quad (3.9)$$

Ovaj raspisani oblik se možda naizgled doima kompliciranijim od 3.7, ali daje pregledan izraz za jednostavnu linearnu interpolaciju.

Pravac $p(x)$ može se zapisati i kao težinska sredina funkcijskih vrijednosti f_0 i f_1

$$p(x) = f_0l_0(x) + f_1l_1(x),$$

gdje su $l_0(x)$ i $l_1(x)$ neke funkcije koje moramo izračunati. Iz općeg uvjeta interpolacije dobiti ćemo jednadžbe

$$p(x_0) = f_0l_0(x_0) + f_1l_1(x_0) = f_0,$$

$$p(x_1) = f_0l_0(x_1) + f_1l_1(x_1) = f_1.$$

Bez nekakvih zadanih pretpostavki ili uvjeta, ne možemo odrediti funkcije $l_0(x)$ i $l_1(x)$. Stoga, pretpostavimo da su polinomi prvog stupnja, ali takvog oblika da ovaj dosad linearni sustav postane dijagonala. Tako za dijagonalne elemente dobivamo

$$l_0(x_0) = 1, l_1(x_1) = 1,$$

a za ostale, odnosno izvandijagonale elemente dobivamo

$$l_1(x_0) = 0, l_0(x_1) = 0.$$

Primjetimo da su funkcije k_0 i k_1 rješenja

$$l_i(x_n) = \delta_{in}, i, n = 0, 1,$$

odnosno, l_i u svim čvorovima osim i -tog mora biti jednak 0, a u i -tom čvoru mora biti jednak 1. Iz tih uvjeta znamo sve nultočke l_i . Stoga funkcije l_0 i l_1 možemo zapisati kao

$$l_0(x) = \frac{x - x_1}{x_0 - x_1}, l_1 = \frac{x - x_0}{x_1 - x_0}.$$

Uvrštavajući te izraze nazad u jednadžbu pravca dobivamo

$$p(x) = f_0 \frac{x - x_1}{x_0 - x_1} + f_1 \frac{x - x_0}{x_1 - x_0}. \quad (3.10)$$

Ovaj izraz je osnova **Lagrangeovog oblika interpolacijskog polinoma**.

4. Lagrangeov oblik interpolacijskog polinoma

Lagrangeov oblik interpolacijskog polinoma, odnosno **Lagrangeova interpolacija**, je takav oblik interpolacije gdje tražena funkcija (odnosno polinom) mora interpolirati samo funkcijske vrijednosti zadane funkcije [1].

Recimo da moramo pronaći polinom $L(x)$ stupnja n , te da taj polinom mora presijecati os x u n točaka, te u nekoj točki i dostići vrijednost 1, odnosno:

$$L_i(x_j) = \begin{cases} 1, & j = i \\ 0, & j \neq i \end{cases}, \quad i, j = 0, 1, \dots, n. \quad (4.1)$$

Pošto polinom L_i mora imati vrijednost 0 u točkama $x_0, x_1, \dots, x_{i-1}, x_{i+1}, \dots, x_n$, raspisujemo

$$L_i = C_i(x - x_0)\dots(x - x_{i-1})(x - x_{i+1})\dots(x - x_n), \quad (4.2)$$

gdje je C_i konstanta koju izvodimo iz uvjeta da polinom L_i mora biti 1 u točki x_i , odnosno

$$C_i = \frac{1}{(x_i - x_0)(x_i - x_1)\dots(x_i - x_{i-1})(x_i - x_{i+1})\dots(x_i - x_n)}. \quad (4.3)$$

Uvrštavajući 4.3 u 4.2 dobijemo

$$L_i(x) = \frac{(x - x_0)\dots(x - x_{i-1})(x - x_{i+1})\dots(x - x_n)}{(x_i - x_0)\dots(x_i - x_{i-1})(x_i - x_{i+1})\dots(x_i - x_n)}, \quad (4.4)$$

što je traženi polinom.

Ovaj polinom će u točki x_i imati vrijednost 1, a u svim ostalim točkama vrijednost 0. Ali što ako želimo da poprimi neku drugu vrijednost? Onda polinom p_i pomnožimo s željenom vrijednosti. Ako te vrijednosti predočimo kao y koordinate nekog skupa od n točaka, polinom možemo zapisati kao

$$P_n = \sum_{i=0}^n y_i \frac{(x - x_0)\dots(x - x_{i-1})(x - x_{i+1})\dots(x - x_n)}{(x_i - x_0)\dots(x_i - x_{i-1})(x_i - x_{i+1})\dots(x_i - x_n)}, \quad (4.5)$$

odnosno

$$P_n(x) = \sum_{i=0}^n y_i l_i(x). \quad (4.6)$$

Polinom P_n zovemo **Lagrangeov oblik interpolacijskog polinoma**. Isti polinom može se napisati na još jedan praktičan način. Definirajmo

$$\omega(x) = \prod_{i=0}^n (x - x_i), \quad (4.7)$$

gdje je $\omega(x)$ produkt svih članova $(x - x_i)$ na rasponu od 0 do n . Stoga izraz 4.4 možemo skratiti i zapisati kao

$$L_i(x) = \frac{\omega(x)}{(x - x_i)\omega_i(x_i)}. \quad (4.8)$$

Uvrštavanjem 4.8 u 4.6 dobivamo

$$P_n(x) = \omega(x) \sum_{i=0}^n \frac{y_i}{(x - x_i)\omega_i(x_i)}. \quad (4.9)$$

Bitno je uočiti da

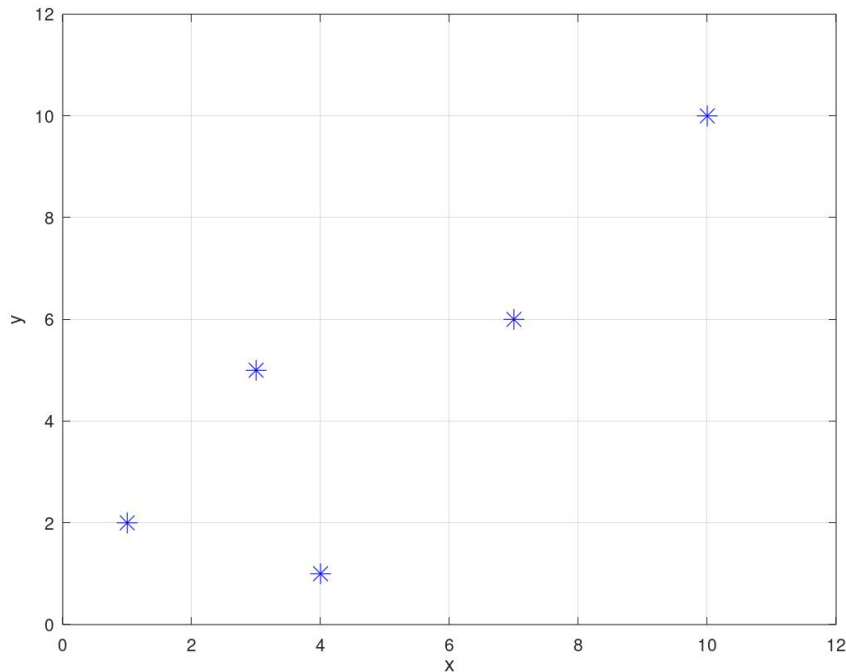
$$\omega_i(x_i) = \omega'(x_i), \quad (4.10)$$

i primjetiti kako nam ova jednakost skraćuje izračun, jer za svaki novi x moramo izračunati samo $\omega(x)$ i $(x - x_i)$, a $\omega_i(x_i) = \omega'(x_i)$ izračunamo samo jednom, jer ne ovisi o x . Uzimajući u obzir 4.10, puni izraz može se zapisati kao

$$P_n(x) = \omega(x) \sum_{i=0}^n \frac{y_i}{(x - x_i)\omega'(x_i)}. \quad (4.11)$$

Ova se forma može koristiti za računanje vrijednosti polinoma u točki $x \neq x_i, i = 0, \dots, n$. Ukupan broj potrebnih izračuna je proporcionalan s n^2 , a za računanje u svakom novom x -u, trebamo još reda veličine n operacija. Ipak, u praksi se ne koristi Lagrangeov oblik, već malo praktičniji Newtonov oblik interpolacijskog polinoma. No o tom obliku ćemo detaljnije u sljedećem poglavlju. Lagrangeov oblik češće se koristi u teoretske svrhe. [3]

Primjer 4.1. *Odredimo interpolacijski polinom čiji graf prolazi sljedećim točkama:*



Slika 4.1. Zadane točke

T_0	(1,2)
T_1	(3,5)
T_2	(4,1)
T_3	(7,6)
T_4	(10,10)

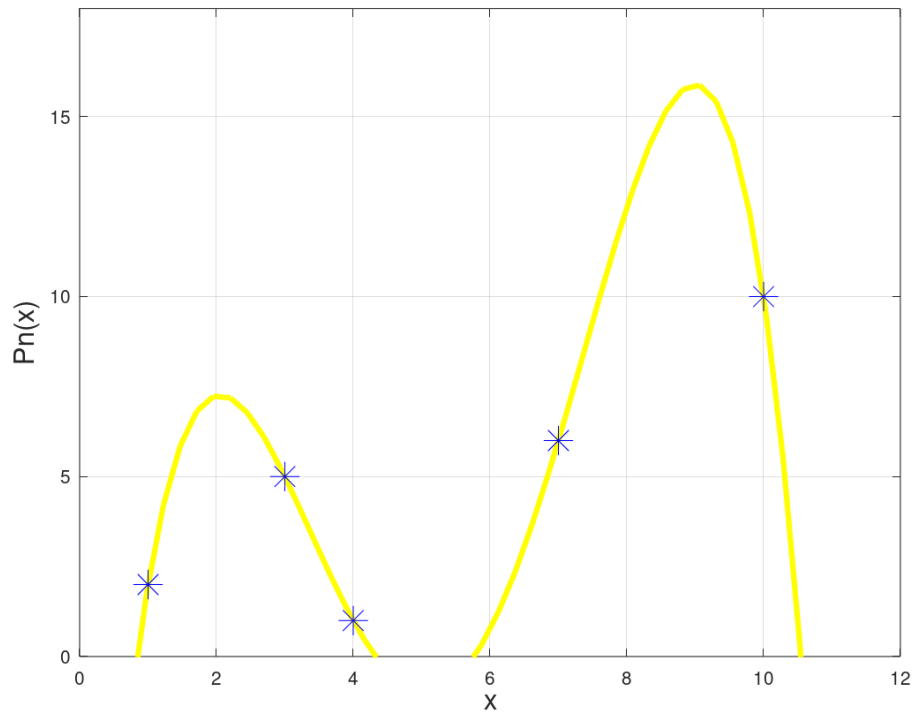
Prema (4.6) raspisujemo sve čvorove interpolacije, odnosno dane točke grafa

$$\begin{aligned}
 P_n(x) = & \frac{(x-3)(x-4)(x-7)(x-10)}{(1-3)(1-4)(1-7)(1-10)} 2 + \frac{(x-1)(x-4)(x-7)(x-10)}{(3-1)(3-4)(3-7)(3-10)} 5 \\
 & + \frac{(x-1)(x-3)(x-7)(x-10)}{(4-1)(4-3)(4-7)(4-10)} 1 + \frac{(x-1)(x-3)(x-4)(x-10)}{(7-1)(7-3)(7-4)(7-10)} 6 \\
 & + \frac{(x-1)(x-3)(x-4)(x-7)}{(10-1)(10-3)(10-4)(10-7)} 10,
 \end{aligned} \tag{4.12}$$

nakon sređivanja dobivamo

$$P_n = -x^4 \frac{379}{4536} + x^3 \frac{1357}{756} - x^2 \frac{18799}{1512} + x \frac{70853}{2268} - \frac{500}{27}. \tag{4.13}$$

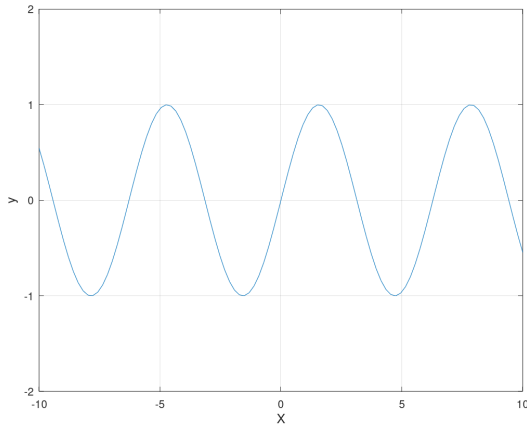
Izračunati Lagrangeov polinom P_n u ovisnosti o x prikazan je na slici 4.2



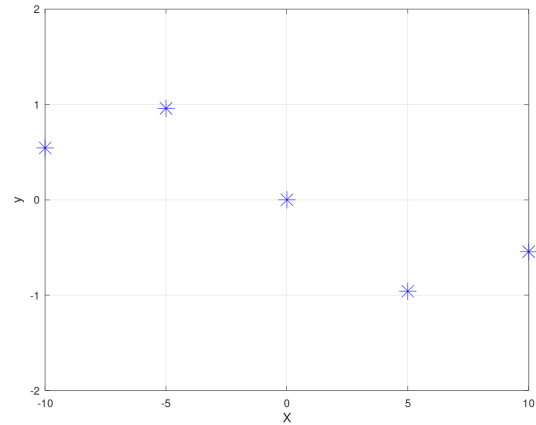
Slika 4.2. Polinom P_n i pripadni čvorovi interpolacije

Vidimo sa slike 4.2 da je Lagrangeov polinom koji smo izračunali u ranijem koraku uvrštavajući zadane čvorove interpolacije glatko i neprekidno povezoao sve točke grafa. Dodavanjem, odnosno uvrštavanjem dodatnih točki u skup čvorova T_n morali bismo nanovo računati cijeli Lagrangeov interpolacijski polinom. U slijedećem primjeru promotriti ćemo proces dodavanja novoizmjerenih podataka i dotičnih komplikacija pri računanju polinoma nanovo za svaki dodatan čvor.

Primjer 4.2. Pri interpoliranju, za dobivanje što točnije rekonstrukcije početne funkcije bitno je, uz to koju metodu koristimo, koliko imamo poznatih podataka odnosno čvorova interpolacije. Za ovaj primjer uzeti ćemo funkciju $\sin(x)$, te je pokušati rekonstruirati pomoću Lagrangeovog polinoma s progresivno više poznatih čvorova.



Slika 4.3. $\sin(x)$



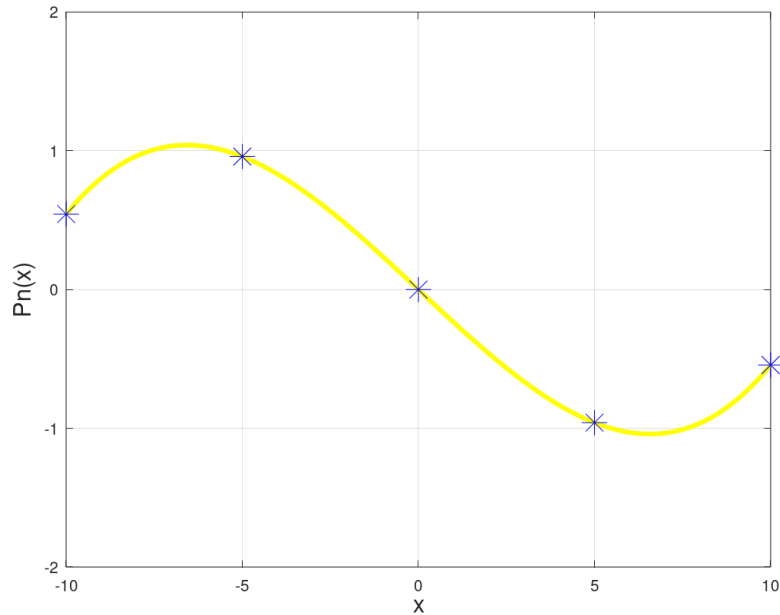
Slika 4.4. Čvorovi uzorkovani s periodom 5

Na slici 4.3 vidimo funkciju $\sin(x)$ ograničenu na interval $x = [-10, 10]$. Za dobivanje čvorova interpolacije možemo periodično uzorkovati danu funkciju, recimo po cijelom intervalu s periodom od 5, odnosno uzamemo čvorove interpolacije na $x = [-10, -5, 0, 5, 10]$, vidljive na slici 4.4.

Za interpoliranje odabranih čvorova koristiti ćemo Lagrangeov polinom iz 4.6. Kao y vrijednosti uvrstiti ćemo $\sin(x)$ vrijednosti za $x = [-10, -5, 0, 5, 10]$

$$P_{1n}(x) = \sum_{i=0}^n y_i l_i = \frac{13}{7097}x^3 - \frac{1028}{4327}x = 0.001831x^3 - 0.237578x. \quad (4.14)$$

Prikažimo izračunati polinom na grafu u ovisnosti o x



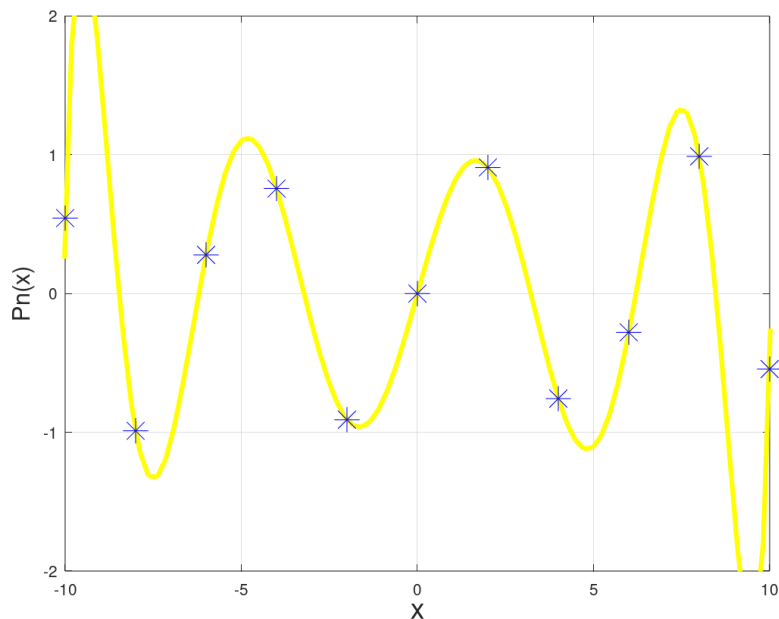
Slika 4.5. Polinom P_{1n} i pripadnih 5 čvorova

Polinom prolazi kroz sve čvorove, ali usporedbom sa slikom 4.3, ako zanemarimo periodičnost funkcije $\sin(x)$, vidimo da graf sa slike 4.5 ne liči na početnu zadanu funkciju. Polinom možemo približiti početnoj funkciji povećanjem broja čvorova.

Uzmimo sad čvor interpolacije svakih $x = 2$, odnosno $x = [-10, -8, -6, -4, -2, 0, 2, 4, 6, 8, 10]$. Svakim dodavanjem novih podataka, odnosno čvorova interpolacije, zanemarujemo ranije izračunat polinom i računamo novi polinom višeg stupnja. Uvrštavanjem točaka koje odgovaraju skupu $T_n = [x_n, \sin(x_n)]$ u izraz 4.6 dobivamo:

$$\begin{aligned}
 P_n(x) = & 3.14 \cdot 10^{-7}x^9 + 4.24 \cdot 10^{-22}x^8 - 6.98 \cdot 10^{-5}x^7 - 5.42 \cdot 10^{-20}x^6 \\
 & + 0.00506x^5 + 1.73 \cdot 10^{-18}x^4 - 0.133x^3 - 6.94 \cdot 10^{-17}x^2 + 0.91x.
 \end{aligned}
 \tag{4.15}$$

Pogledajmo grafički prikaz interpolirane funkcije iscrtane preko čvorova:



Slika 4.6. Polinom P_{2n} i pripadnih 11 čvorova

Usporedimo dva izračunata polinoma. U odnosu na prvi polinom, polinom P_{2n} je preciznije replicirao funkciju na intervalu $x = [-5, 5]$; npr. uspješno je prikazao da se jedna perioda $\sin(x)$ funkcije izvrši na intervalu od $[0, 2\pi]$, za razliku od polinoma P_{n1} , koji je periodu udvostručio. Ali, zbog potrebe za prolaskom kroz veći broj čvorova, u graničnim područjima intervala funkcija postaje vrlo nepregledna i odlazi u ekstreme. Izvan čvorova interpolacije funkcija postaje gotovo vertikalna, ali uzimajući u kontekst da Lagrangeova interpolacija nema nikakav sustav za predviđanje vrijednosti izvan intervala, odnosno nema sustav za **ekstrapolaciju**, ne možemo očekivati točnost izvan zadanih točaka.

Iz prijašnjeg primjera vidljivo je da povećanjem broja čvorova možemo preciznije rekonstruirati početnu funkciju. Zadatak inženjera u ovom slučaju, je odrediti broj čvorova (odnosno intervale uzorkovanja) kako bi omjer korištene računalne memorije i kvalitete rekonstrukcije signala bio zadovoljavajući.

5. Newtonov oblik interpolacijskog polinoma

Lagrangeov oblik interpolacijskog polinoma, kao što smo prikazali u primjerima prethodnog poglavlja, nepogodan je kada želimo povećati stupanj interpolacijskog polinoma jer pri dodavanju svakog čvora polinom moramo računati nanovo. To stvara nepoželjnu količinu posla kada želimo dodavati samo par čvorova kako bi smo poboljšali aproksimaciju, ili kada želimo izuzeti par čvorova kako bi smo pojednostavili polinom. Ti procesi su bitno pojednostavljeni pri korištenju **Newtonovog oblika interpolacijskog polinoma**.

Recimo da je p_{n-1} polinom koji interpolira neku funkciju f u točkama $x_i, i = 0, \dots, n-1$. Uz taj polinom, recimo da je p_n polinom koji istu funkciju f interpolira još i u točki x_n . Taj polinom možemo zapisati na sljedeći način:

$$p_n(x) = p_{n-1} + c(x), \quad (5.1)$$

gdje $c(x)$ predstavlja korekciju između dva stupnja interpolacijskog polinoma. Za tu korekciju mora vrijediti

$$c(x_k) = p_n(x_k) - p_{n-1}(x_k) = f(x_k) - f(x_k) = 0, i = 0, \dots, n-1.$$

Vidljivo je da su x_i nultočke, odnosno razumijemo da u poznatim čvorovima nema potrebe za korekcijom. Stoga izraz možemo zapisati kao

$$c(x) = a_n(x - x_0)\dots(x - x_{n-1}). \quad (5.2)$$

Uvrštavajući 5.2 nazad u 5.1, te poštivajući uvjet interpolacije $f(x_n) = p_n(x_n)$ dobivamo

$$f(x_n) = p_n(x_n) = p_{n-1}(x_n) + a_n(x_n - x_0)\dots(x_n - x_{n-1}), \quad (5.3)$$

te iz tog izraza lako računamo koeficijent a_n polinoma $c(x)$

$$a_n = \frac{p_n(x_n) - p_{n-1}(x_n)}{(x_n - x_0)\dots(x_n - x_{n-1})}. \quad (5.4)$$

Koeficijent a_n je funkcija zadanih čvorova x_0, \dots, x_n i u literaturi se naziva n -ta podijeljena razlika [1]. U literaturi se označuje s

$$a_n = f[x_0, x, \dots, x_n],$$

te ćemo je u tom obliku zapisati u izrazu 5.3

$$p_n(x_n) = p_{n-1}(x_n) + f[x_0, x, \dots, x_n](x_n - x_0) \dots (x_n - x_{n-1}). \quad (5.5)$$

Izraz iznad je rekurzivna formula za dobivanje interpolacijskog polinoma za stupanj većeg od prethodnog [1]. Kako bi bolje opisali koeficijent a_n , podsjetimo se Lagrangea. Promotrimo ranije spomenutu relaciju 4.11. Uočimo da se a_n nalazi uz vodeću potenciju x^2 .

$$a_n = f[x_0, x, \dots, x_n] = \sum_{i=0}^n \frac{y(x_i)}{\omega'(x_i)}. \quad (5.6)$$

Iz formule 5.6 možemo izvući svojstva podijeljenih razlika. Primjetimo da, za razliku od Lagrangea, poredak čvorova u dotičnoj jednakosti nije bitan. Druga korisna formula je formula za rekurzivno računanje podijeljenih razlika [3]

$$f[x_0, \dots, x_n] = \frac{f[x_1, \dots, x_n] - f[x_0, \dots, x_{n-1}]}{x_n - x_0}. \quad (5.7)$$

Izvedimo gornju formulu. Vrijedi

$$\begin{aligned} f[x_1, \dots, x_n] &= \sum_{i=1}^n \frac{f(x_i)}{(x_i - x_1) \dots (x_i - x_{i-1})(x_i - x_{i+1}) \dots (x_i - x_n)} \\ &\quad + \sum_{i=1}^{n-1} \frac{f(x_i)(x_i - x_0)}{(x_i - x_0) \dots (x_i - x_{i-1})(x_i - x_{i+1}) \dots (x_i - x_n)} \\ &\quad + \frac{f(x_n)(x_n - x_0)}{(x_n - x_0) \dots (x_n - x_{n-1})}, \end{aligned} \quad (5.8)$$

odnosno

$$\begin{aligned} f[x_0, \dots, x_{n-1}] &= \sum_{i=0}^{n-1} \frac{f(x_i)}{(x_i - x_0) \dots (x_i - x_{i-1})(x_i - x_{i+1}) \dots (x_i - x_{n-1})} \\ &\quad + \sum_{i=1}^{n-1} \frac{f(x_i)(x_i - x_n)}{(x_i - x_0) \dots (x_i - x_{i-1})(x_i - x_{i+1}) \dots (x_i - x_n)} \\ &\quad - \frac{f(x_0)(x_n - x_0)}{(x_0 - x_1) \dots (x_0 - x_n)}. \end{aligned} \quad (5.9)$$

Oduzimanjem 5.8 od 5.9 dobivamo

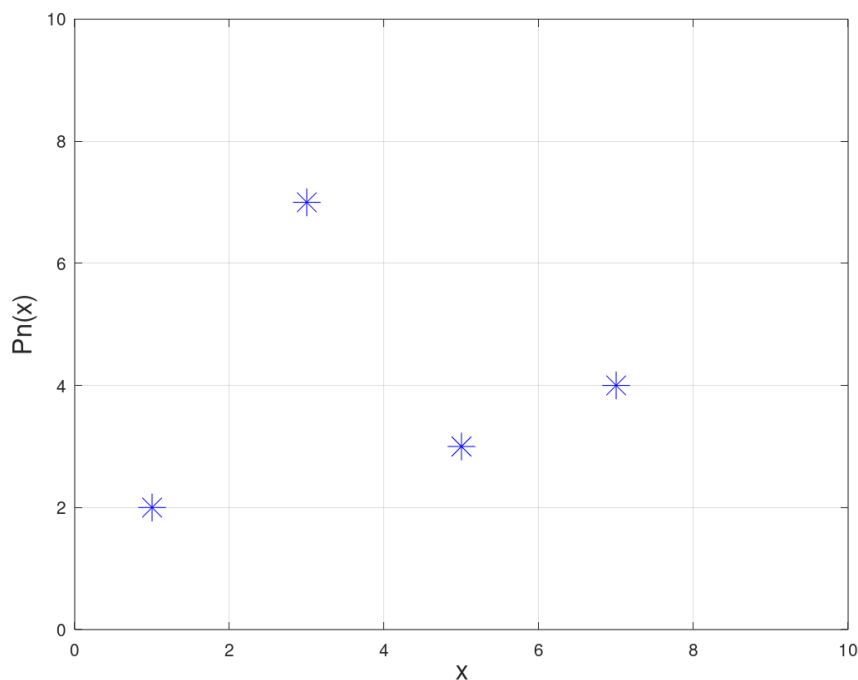
$$\begin{aligned}
f[x_1, \dots, x_n] - f[x_0, \dots, x_{n-1}] &= \sum_{i=1}^{n-1} \frac{f(x_i)(x_n - x_0)}{(x_i - x_0)\dots(x_i - x_{i-1})(x_i - x_{i+1})\dots(x_i - x_n)} \\
&+ \frac{f(x_n)(x_n - x_0)}{(x_n - x_0)\dots(x_n - x_{n-1})} + \frac{f(x_0)(x_n - x_0)}{(x_0 - x_1)\dots(x_0 - x_n)} \quad (5.10) \\
&= (x_n - x_0) \sum_{i=0}^n \frac{f(x_k)}{\omega'(x_k)} = (x_n - x_0)f[x_0, \dots, x_n],
\end{aligned}$$

čime smo dokazali danu formulu. Ova rekurzivna formula se često koristi kao definicija **podijeljenih razlika**. [3]

Promotrimo kako rekurzija počinje kod podijeljenih razlika. Ako znamo da neka konstanta k prolazi nekom točkom $T = (x_0, f(x_0))$, tu konstantu možemo definirati kao polinom stupnja 0. To znači da je koeficijent $a_0 = f[x_0] = f(x_0)$. Uzimajući to u obzir, dobivamo konačan oblik Newtonovog interpolacijskog polinoma

$$\begin{aligned}
p_n(x) &= f[x_0] + (x - x_0)f[x_0, x_1] + (x - x_0)(x - x_1)f[x_0, x_1, x_2] + \dots \\
&\dots + (x - x_0)\dots(x - x_{n-1})f[x_0, x_1, \dots, x_n]. \quad (5.11)
\end{aligned}$$

Primjer 5.1. Odredimo Newtonov interpolacijski polinom koji prolazi sljedećim točkama:



Slika 5.1. Zadane točke

T_0	(1,2)
T_1	(3,7)
T_2	(5,3)
T_3	(7,4)

Koristeći izraz 5.11, uvrstimo čvorove zadane u tablici. Član $f[x_0]$ je samo $P_n(x)$ vrijednost prvog čvora, odnosno točke T_0 . Preostale podijeljene razlike potrebno je izračunati.

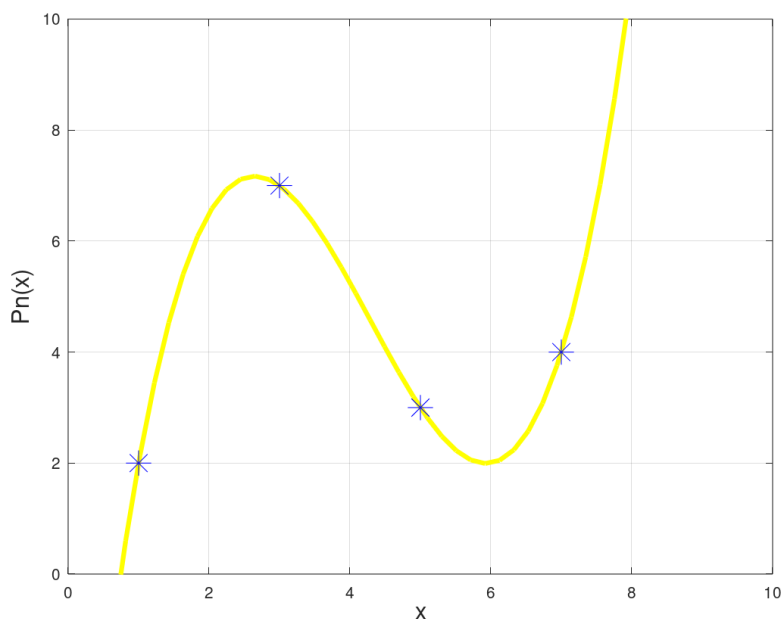
$$\begin{aligned}
 f[x_0, x_1] &= \frac{7-2}{3-1} = \frac{5}{2}, f[x_1, x_2] = \frac{3-7}{5-3} = -2, f[x_2, x_3] = \frac{4-3}{7-5} = \frac{1}{2}, \\
 f[x_0, x_1, x_2] &= \frac{-2 - \frac{5}{2}}{5-1} = -\frac{9}{8}, f[x_1, x_2, x_3] = \frac{\frac{1}{2} + 2}{7-3} = \frac{5}{8}, \\
 f[x_0, \dots, x_3] &= \frac{\frac{5}{8} + \frac{9}{8}}{7-1} = \frac{7}{24}.
 \end{aligned} \tag{5.12}$$

Raspišimo izračunate članove u polinom

$$p_n(x) = 2 + (x-1)\frac{5}{2} + (x-1)(x-3)\left(-\frac{9}{8}\right) + (x-1)(x-3)(x-7)\frac{7}{24}, \tag{5.13}$$

odnosno, nakon sređivanja

$$p_n(x) = \frac{7}{24}x^3 - \frac{13}{3}x^2 + \frac{385}{24}x - 10. \tag{5.14}$$



Slika 5.2. Newtonov interpolacijski polinom p_n i pripadni čvorovi

Ispitajmo ranije spomenutu prednost Newtonovog interpolacijskog polinoma - bitno pojednostavljenu računicu pri dodavanju novih čvorova. Dodajmo još dva čvora nakon prethodno zadanih - $T_4 = (10, 5)$ i $T_5 = (12, 10)$

Dodavanjem dva čvora sveukupno imamo šest čvorova interpolacije, pa će novi polinom $p_n(x)$ biti polinom 5-tog stupnja. Kako su dva nova polinoma niže na osi apscisa, možemo iskoristiti prijašnji polinom, te samo izračunati i nadodati nove podijeljene razlike.

Ponavljanjem postupka 5.12, nove podijeljene razlike su

$$f[x_0, \dots, x_4] = -\frac{3}{70}$$

$$f[x_0, \dots, x_5] = \frac{17}{3080}.$$

Po izrazu 5.11, nove podijeljene razlike samo nadodamo na ranije izračunati polinom, te ih pomnožimo s $(x - x_0)\dots(x - x_{n-2})$, odnosno $(x - x_0)\dots(x - x_{n-1})$.

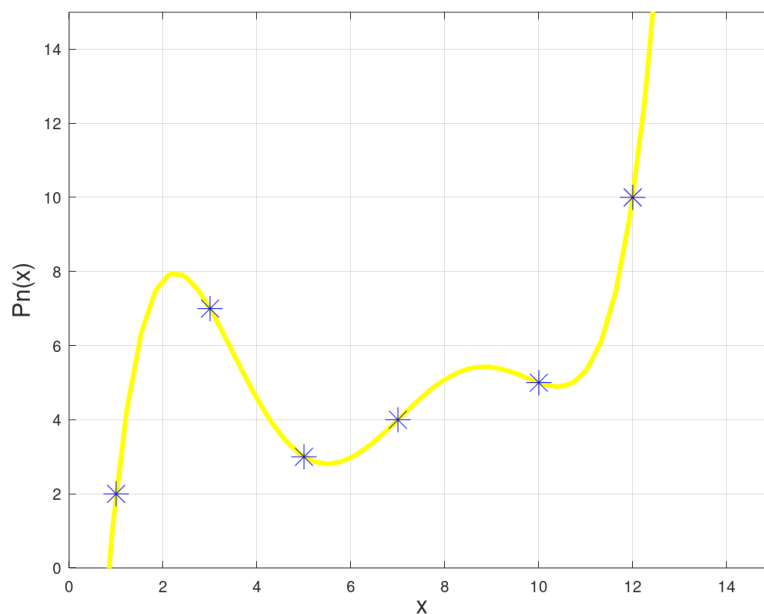
Raspišimo novi polinom

$$p_n(x) = 2 + (x - 1)\frac{5}{2} + (x - 1)(x - 3)\left(-\frac{9}{8}\right) + (x - 1)(x - 3)(x - 7)\frac{7}{24}$$

$$+ (x - 7)(x - 5)(x - 3)(x - 1)\left(-\frac{30}{70}\right) + (x - 10)(x - 7)(x - 5)(x - 3)(x - 1)\frac{17}{3080}.$$

Vidimo da je novi polinom do člana $f[x_0, \dots, x_4]$ jednak prijašnjem. Nakon sređivanja dobivamo

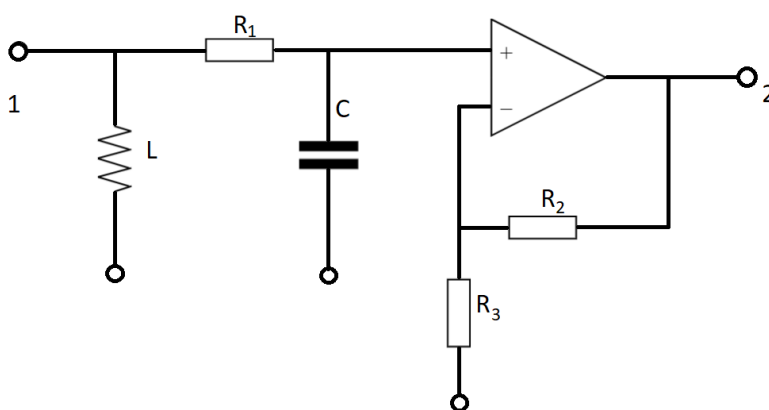
$$p_n(x) = \frac{17}{3080}x^5 - \frac{41}{220}x^4 + \frac{21577}{9240}x^3 - \frac{20257}{1540}x^2 + \frac{72869}{2310}x - \frac{204}{11}. \quad (5.15)$$



Slika 5.3. Novi Newtonov interpolacijski polinom p_n i dodatni pripadni čvorovi

Raniji primjeri su ispitali ponašanje Newtonovog interpolacijskog polinoma na arbitrarnim primjerima nasumično zadanih točaka, bez poznavanja povezuje li dotične točke neka stvarna relacija između varijable x i veličine $P_n(x)$. U sljedećem primjeru promotriti ćemo ponašanje i Lagrangeovog i Newtonovog interpolacijskog polinoma na primjeru rekonstrukcije amplitudno-frekvencijske karakteristike električnog filtra.

Primjer 5.2. Usporedimo Lagrangeov i Newtonov interpolacijski polinom na primjeru iz struke. Imamo električnu mrežu, odnosno filter, kao na slici 5.4. Moramo odrediti amplitudno-frekvencijsku karakteristiku spoja, tj. vrijednost prijenosne funkcije koja predstavlja odnos amplitude napona na izlaznom priključku 2 u odnosu na amplitudu napona na priključku 1 pri promjeni frekvencije ulaznog napona. Voltmetrom smo očitali vrijednosti prijenosne funkcije pri 5 različitih frekvencija, prikazanih u tablici.



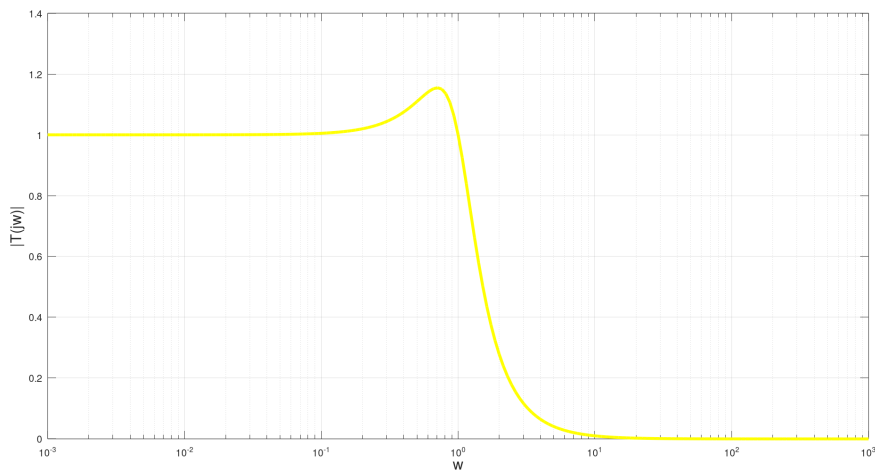
Slika 5.4. Zadana shema

ω	$ T(j\omega) $
10^{-2}	1
10^{-1}	1,005
10^0	1,16
10^1	0,004
10^2	0

Tablica 5.1. Odnosi frekvencije i amplitude prijenosne funkcije

Kako bi smo imali izraz za referencu raspisati ćemo mrežu po drugom Kirchhoffovom zakonu, koji glasi da je algebarski zbroj svih napona u nekoj zatvorenoj električnoj petlji jednak nuli. Pretpostavimo da su vrijednosti elemenata u mreži normirane na takav način da, nakon prebacivanja u frekvencijsku domenu preko Laplaceove transformacije, dobijemo sljedeći izraz kao odnos napona U_1 na priključku 1 i napona U_2 na izlaznom priključku 2

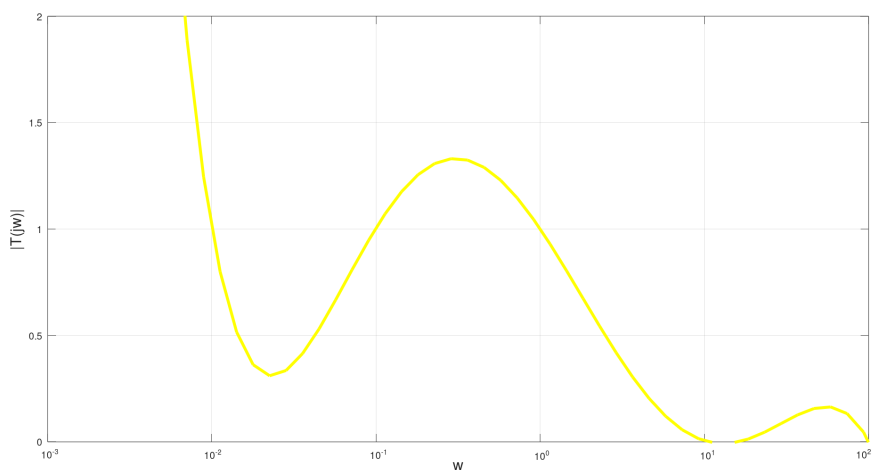
$$|T(j\omega)| = \left| \frac{U_2}{U_1} \right| = \frac{1}{\sqrt{(1 - \omega^2)^2 + \omega^2}}. \quad (5.16)$$



Slika 5.5. Prikaz izračunate prijenosne funkcije u frekvencijskoj domeni

Na slici iznad vidimo odnos apsolutne vrijednosti omjera napona na izlazu i ulazu sklopa u ovisnosti o frekvenciji ulaznog napona. Normiranjem frekvencije, odnosno linearizacijom ovisnosti napona o frekvenciji, možemo i uzorkovati amplitudu napona, te onda parove frekvencije i omjera amplituda uvrštavamo u Lagrangeov interpolacijski polinom. Nakon izračuna dobivamo izraz

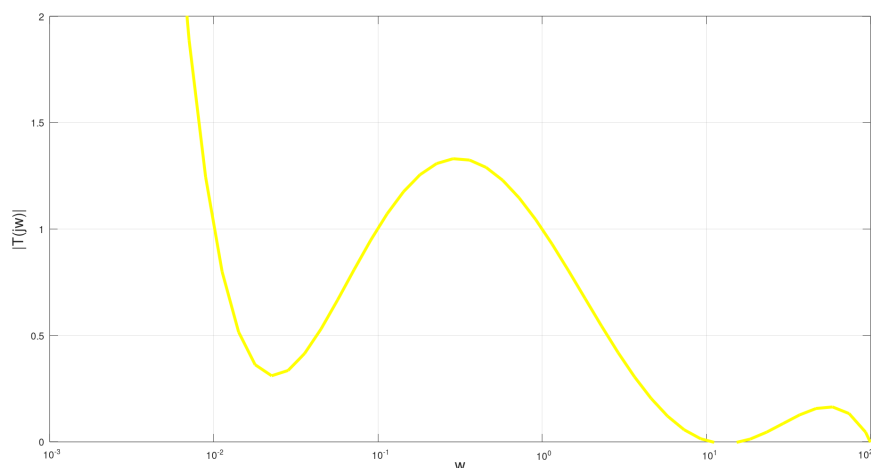
$$|T(j\omega)| = -0,1398\omega^5 + 1,5217\omega^4 - 5,7984\omega^3 + 8,83531x^2 - 4,41383\omega + 1. \quad (5.17)$$



Slika 5.6. Lagrangeov interpolacijski polinom uzorkovane amplitudno-frekvencijske karakteristike

Učinimo isto s Newtonovim interpolacijskim polinomom

$$|T(j\omega)| = -0,1398\omega^5 + 1,5217\omega^4 - 5,7984\omega^3 + 8,83531x^2 - 4,41383\omega + 1. \quad (5.18)$$



Slika 5.7. Newtonov interpolacijski polinom uzorkovane amplitudno-frekvencijske karakteristike

Vidimo iz izraza 5.17 i 5.18, kao i iz slika 5.6 i 5.7, da su konačna dva polinoma ista. Gledajući slike možemo zaključiti da, neovisno o tome što prolaze kroz sve uzorkovane, odnosno zadane čvorove interpolacije, krivulje ne predstavljaju stvarnu A-f karakteristiku na adekvatan način. Odnosno, prikazuju vrijednost amplitude samo u konkretnim vrijednostima frekvencije, te ne prikazuju ponašanje amplitude pri određenom pojasu frekvencija, kao što je znatno bolje prikazano na slici 5.5.

Ne bi li smo "poboljšali" karakteristiku, odnosno približili je originalnoj izračunatoj krivulji, moramo dodati još podataka. To najbolje možemo postići uključivanjem još čvorova u polinome, što je bitnije pojednostavljeno s Newtonovim interpolacijskim polinomom.

6. Spline interpolacija

U prijašnjim poglavljima definirali smo dvije metode polinomne interpolacije, odnosno dva interpolacijska polinoma - Lagrangeov i Newtonov interpolacijski polinom. Kroz primjere smo vidjeli da obje metode kroz različit postupak u konačnici omogućuju računanje glatkih polinoma koji prolaze kroz sve zadane čvorove. Međutim, pri suočavanju s realnim inženjerskim problemima, može se dogoditi da se kroz mjerenja ili uzorkovanja susrećemo s podacima koji su na određenom skupu gusto ili nepraktično raspoređeni. U takvoj situaciji računanje s ranije spomenutim interpolacijskim polinomima može izazvati oscilacije u interpoliranoj funkciji. Uz to, interpolacija velikog broja čvorova polinomom visokog stupnja može biti računalno zahtjevna, pa s te strane i inženjerski nepraktična.

Kako bi riješili ovu problematiku prekomjerenih oscilacija i prevelikog broja gusto zbijenih podataka, koristiti ćemo spline interpolaciju. Takav tip interpolacije upotrebljava glatke polinome niskog stupnja koji su prispojeni tako da se minimiziraju oscilacije. Odnosno, možemo to zamisliti kao konstrukciju novih i praktičnijih čvorova na temelju zadanih, te povezivanje tih novih čvorova polinomom.

6.1. Linerani spline

Recimo da je $f : [a, b] \rightarrow \mathbb{R}$ neprekidna funkcija, te da su nam poznate njene vrijednosti y_0, y_1, \dots, y_n u $(n + 1)$ čvorova. Recimo da su čvorovi na skupu $[a, b]$ raspoređeni tako da vrijedi $a = x_0 < x_1 < \dots < x_{n-1} < x_n = b$. Funkciju f ćemo interpolirati po dijelovima s linearnom funkcijom $\phi : [a, b]$, tako da je neprekidna po cijelom skupu. Za $i = 1, \dots, n$ definirajmo

$$\phi_i(x) = y_{i-1} + \frac{y_i - y_{i-1}}{x_i - x_{i-1}}(x - x_{i-1}), x \in [x_{i-1}, x_i]. \quad (6.1)$$

Pošto je

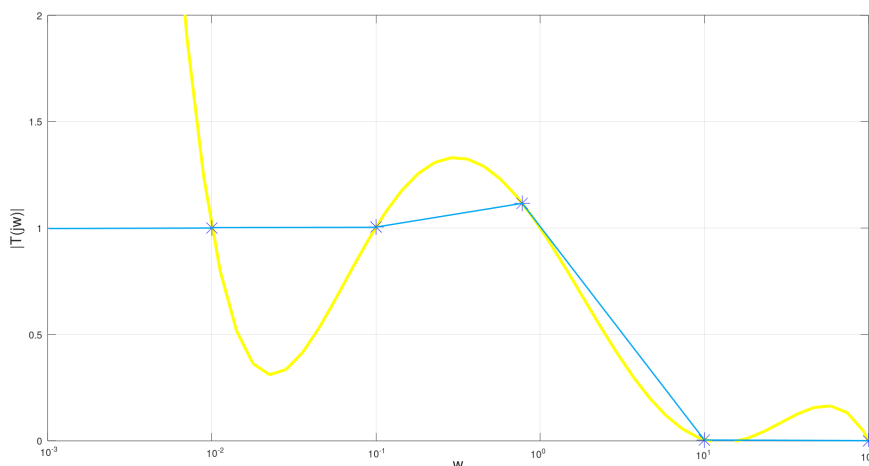
$$\phi_i(x) = y_i = \phi_{i+1}(x_i), i = 1, \dots, n - 1, \quad (6.2)$$

funkcija ϕ neprekidna je na intervalu $[a, b]$, a linearna na svakom podintervalu između dva čvora $[x_{i-1}, x_i]$.

Pogledajmo kako bi linearna spline interpolacija izgledala na podacima iz primjera 5.2. Za potrebu ove usporedbe možemo pretpostaviti da je odnos između frekvencija na grafu linearan. Odnosno, zamjenimo frekvencije $\omega = [10^{-1}, 10^0]$ s $w = [1, 2]$. Uvrštavajući parove čvorova dobivamo jednadžbu pravca koja opisuje funkciju između ta dva čvora, odnosno jedan spline

$$\phi(w) = 1,005 + \frac{1,16 - 1,005}{2 - 1}(x - 1) = 0,155w + 0,85, w \in [10^{-1}, 10^0]. \quad (6.3)$$

Ponavljajući taj postupak za sve parove čvorova na rasponu $[10^{-2}, 10^2]$ dobivamo sljedeću karakteristiku



Slika 6.1. Prikaz linearnog interpolacijskog splinea (plavo) preko Newtonovog i Lagrangeovog interpolacijskog polinoma (žuto) iz primjera 5.2.

Vidimo da je linearni spline povezoao sve čvorove interpolacije, te uspoređujući ga s izračunatom karakteristikom filtra na slici 5.5, vidimo da realnije prikazuje ponašanje filtra pri različitim frekvencijama. No ipak, spline interpolacijom dobivamo familiju pravaca koji dijele jednu zajedničku točku (čvor) sa susjednim pravcem. Kao takve, te funkcije su linearne po svakom podintervalu, što znači da ako je originalna funkcija krivulja da linearni spline nikako ne opisuje kretanje funkcije u području oko čvora. Također, linearne spline funkcije su nederivabilne u čvorovima. Zbog toga nisu pretjerano pogodne za danje računanje. Kako bi dobili funkcije koje zadržavaju podijeljenu prirodu linearnog splinea, ali su glatke po cijeloj dužini, u idućem potpoglavlju proučiti ćemo pojam kubičnog splinea.

6.2. Kubični spline

Kubični spline je metoda interpolacije koja se koristi za stvaranje glatkih interpolacijskih funkcija, odnosno glatko povezivanje podataka. Osnova ideja kubičnog splinea je da se poznati podaci, odnosno čvorovi, povežu slično kao i u prijašnjem slučaju s linearnim splineovima. Ali umjesto funkcija pravaca, funkcija se na manjim segmentima između pojedinih čvorova aproksimira kubičnim polinomima, odnosno polinomima trećeg reda. Kubični splineovi omogućavaju glatko povezivanje susjednih segmenata, čiji je rezultat funkcija koja je kontinuirana, glatka i derivabilna u svim točkama.

Recimo da je $f : [a, b] \rightarrow \mathbb{R}$ neprekidna funkcija, te da poznamo njene vrijednosti y_0, y_1, \dots, y_n u $(n + 1)$ čvorova. Čvorovi su na skupu $[a, b]$ raspoređeni na način da je $a = x_0 < x_1 < \dots < x_{n-1} < x_n = b$. Funkciju f ćemo aproksimirati funkcijom $K : [a, b] \rightarrow \mathbb{R}$, tako da vrijedi

$$K(x) = K_i(x), x \in [x_{i-1}, x_i], i = 1, \dots, n, \quad (6.4)$$

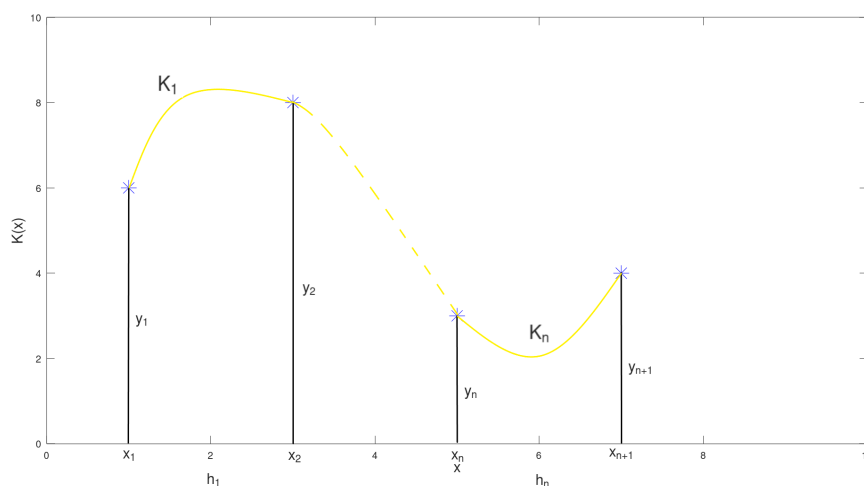
gdje su $K_i, i = 1, \dots, n$, polinomi trećeg reda koje uobičajeno pišemo kao

$$K_i(x) = c_{0,k} + c_{1,k}(x - x_{i-1}) + c_{2,k}(x - x_{i-1})^2 + c_{3,k}(x - x_{i-1})^3, x \in [x_{i-1}, x_i], i = 1, \dots, n, \quad (6.5)$$

gdje su $c_{0-3,k}$ koeficijenti koje moramo odrediti tako da zadovoljavaju sljedeće uvjete

$$\begin{aligned} (i) & K_i(x_{i-1}) = y_{i-1}, i = 1, \dots, n; \\ (ii) & K_i(x_i) = y_i, i = 1, \dots, n; \\ (iii) & K'_i(x_i) = K'_{i+1}(x_i), i = 1, \dots, n - 1; \\ (iv) & K'_a(a) = f'(a), K'_b(b) = f'(b); \\ (v) & K''_i(x_i) = K''_{i+1}(x_i), i = 1, \dots, n - 1; \\ (vi) & K''_1(x_0) = K''_n(x_n) = 0. \end{aligned} \quad (6.6)$$

Uvjeti (i) – (ii) navedeni u 6.6 su potpuni rubni uvjeti i osiguravaju interpolacijske uvjete i neprekidnost funkcije, te se za splineove koji ispunjavaju te uvjete tradicionalno koristi naziv potpuni splineovi. Dok su uvjeti (iii) – (vi) prirodni rubni uvjeti koji osiguravaju derivabilnosti u čvorovima interpolacije, a i samim time i glatkost funkcije u svim točkama uključujući i čvorove. Splineovi koji ispunjavaju te uvjete nazivaju se prirodnim splineovima. Funkcije koje zadovoljavaju sve uvjete (i) – (vi) zovemo prirodni kubični interpolacijski spline [2].



Slika 6.2. Kubični interpolacijski spline

Algoritam za konstruiranje interpolacijskog kubičnog splajna možemo izvesti na dva načina. Prvi način, gdje za nepoznate parametre koje trebamo odrediti uzimamo vrijednosti prve derivacije splinea u čvorovima. U literaturi je najčešće korištena oznaka za te parametre $m_i := K'_i(x_i)$, $i = 0, \dots, n$. Drugi način, u kojem uzimamo vrijednosti druge derivacije splinea u čvorovima kao nepoznate parametre, koristeći globalnu neprekidnost $K_i(x_i)$, uz oznaku $M_i := K''_i(x_i)$, $i = 0, \dots, n$. Ovdje je bitno napomenuti da se ta dva algoritma dosta ravnopravno koriste u praksi, a za ocjenu greške moramo koristiti i oba, pa ćemo izvesti i prvi i drugi način.

Prvi algoritam izvodimo pomoću Hermiteove interpolacije, ali u ovom slučaju ne zadajemo derivacije, već nepoznate derivacije m_i uzimamo kao parametre koje trebamo odrediti tako da postignemo globalnu pripadnost splinea klasi $C^2[a, b]$ [1]. Odnosno, postavljamo $K_i(x_i)$ tako da polinom zadovoljava uvjete $K_i(x_i) = y_i$, $i = 1, \dots, n$, $K'_i(x_i) = m_i$, $i = 1, \dots, n$, gdje su y_i zadane vrijednosti, a m_i nepoznate vrijednosti koje moramo odrediti. Polinom K_i na svakom podintervalu možemo zapisati kao

$$K_i(x) = y_i(1-t)^2(1+2t) + y_{i+1}t^2(3-2t) + m_i h_i t(1-t)^2 - m_{i+1} h_i t^2(1-t), \quad (6.7)$$

gdje je $t = \frac{x-x_i}{h_i}$ za $x \in [x_i, x_{i+1}]$, a $h_i = x_{i+1} - x_i$, $i = 0, \dots, n-1$. Parametre m_i, m_{i+1} trebamo odrediti da dvostruka derivacija $K''_i(x)$ bude neprekidna u čvorovima. Raspišemo li tu derivaciju kao

$$K''_i(x_i) = \frac{y_{i+1} - y_i}{h_i^2}(6 - 12t) + \frac{m_i}{h_i}(6t - 4) + \frac{m_{i+1}}{h_i}(6t - 2), \quad (6.8)$$

iz čega slijedi

$$\begin{aligned} K''_i(x_i + 0) &= 6\frac{y_{i+1} - y_i}{h_i^2} - 4\frac{m_i}{h_i} - 2\frac{m_{i+1}}{h_i}, \\ K''_i(x_i + 0) &= -6\frac{y_i - y_{i-1}}{h_{i-1}^2} + 4\frac{m_i}{h_{i-1}} + 2\frac{m_{i-1}}{h_{i-1}}. \end{aligned} \quad (6.9)$$

Koristeći oznake

$$\begin{aligned} \sigma_i &= \frac{h_{i-1}}{h_{i-1} + h_i}, \lambda_i = 1 - \sigma_i, \\ c_i &= 3\left(\sigma_i \frac{y_{i+1} - y_i}{h_i} + \lambda_i \frac{y_i - y_{i-1}}{h_{i-1}}\right), \end{aligned} \quad (6.10)$$

možemo uvjete neprekidnosti $K''_i(x_i)$, $i = 1, \dots, n-1$ zapisati kao

$$\lambda_i m_{i-1} + 2m_i + \sigma_i m_{i+1} = c_i, \quad i = 1, \dots, n-1. \quad (6.11)$$

Ovime dobivamo $n - 1$ jednadžbi s $n + 1$ nepoznanica m_i . To znači da nam nedostaju dvije jednadžbe. Njih možemo dobiti iz uvjeta raspisanih u 6.6, te uz te uvjete možemo raspisati linearni sustav jednadžbi oblika

$$\begin{aligned} 2m_0 + \sigma_0 * m_1 &= c_0*, \\ \lambda_i m_{i-1} + 2m_i + \sigma_i m_{i+1} &= c_i, i = 1, \dots, n - 1, \\ \lambda_n m_{n-1} + 2m_n &= c_n * . \end{aligned} \quad (6.12)$$

gdje su σ_0* , c_0* , λ_n* i c_n* koeficijenti koje određujemo zavisno o rubnim uvjetima raspisanim u 6.6. Za uvjete (vi) uvažavamo jednakosti

$$\sigma_0* = \lambda_n* = 0, c_0* = 2y'_0, c_n* = 2y_n, \quad (6.13)$$

gdje su y_0 i y_n vrijednosti funkcije f na granicama intervala $[a, b]$, odnosno u točkama $x_0 = a$ i $x_n = b$. Dok za uvjete (v) vrijedi

$$\begin{aligned} \sigma_0* = \lambda_n* = 1, c_0* &= 3\frac{y_1 - y_0}{h_0} - \frac{h_0}{2}f''(x_0), \\ c_n* &= 3\frac{y_n - y_{n-1}}{h_{n-1}} + \frac{h_n - 1}{2}f''(x_n). \end{aligned} \quad (6.14)$$

Za slučaj kad je funkcija f periodička, vrijedi da je $y_0 = y_n$, te da je $m_0 = m_n$ radi periodičkog uvjeta na prvu derivaciju (iii). Kako bismo lakše izrazili periodičnost pri drugoj derivaciji, možemo produljiti funkciju, tj. povećati broj čvorova dodavanjem čvora x_{n+1} , uz uvjet da je $x_{n+1} - x_n = x_1 - x_0$, odnosno da je $h_0 = h_n$. S obzirom na pretpostavku periodičnosti, trebamo uvjetovati $y_{n+1} = y_1$ i $m_{n+1} = m_1$. Uvažavajući tu pretpostavku, uvjet periodičnosti druge derivacije postaje istovjetan uvjetu kontinuiteta druge derivacije u točki x_n , odnosno jednadžbi 6.11 kada je $i = n$. Uvažavajući pretpostavke

$$y_0 = y_n, y_{n+1} = y_1, m_0 = m_n, m_1 = m_{n+1}, h_0 = h_n, \quad (6.15)$$

možemo raspisati sustav s n brojem jednadžbi

$$\begin{aligned} 2m_i + \sigma_i m_{i+1} + m_{i+1} \lambda_i &= c_i, i = 1, \dots, n - 1, \\ 2m_n + \sigma_n m_1 + m_{n-1} \lambda_n &= c_n. \end{aligned} \quad (6.16)$$

Raspisani sustav za svaku vrijednost c_n možemo uvrstiti nazad u izraz 6.5.

Drugi algoritam raspisujemo na svakom pojedinačnom podintervalu $[x_i, x_{i+1}]$ koji pripada skupu $f : [a, b]$, isto kao i kod prvog algoritma, ali umjesto koeficijenta m_i , moramo izračunati koeficijente $M_i = K_i''(x_i)$, $i = 0, \dots, n$. Polinom K_i određuje se na temelju uvjeta interpolacije funkcije (i) i (ii), te iz uvjeta druge derivacije u rubovima funkcije

$$\begin{aligned} K_i(x_i) &= y_i, K_{i+1}(x_{i+1}) = y_{i+1}, \\ K_i''(x_i) &= M_i, K_{i+1}''(x_{i+1}) = M_{i+1}, \end{aligned} \quad (6.17)$$

Polinom K također možemo raspisati kao linearnu interpolaciju za poznate vrijednosti funkcije s nadodanom korekcijom koja uključuje koeficijente druge derivacije M_i i M_{i+1}

$$K(x) = y_i(1-t) + y_{i+1}t - t \frac{h_i^2}{6} (1-t)(M_i(2-t) + M_{i+1}(1+t)), \quad (6.18)$$

gdje su t i h_i iste vrijednosti kao i u prijašnjem zapisu 6.7. Derivacijom izraza 6.18 dobivamo

$$\begin{aligned} K'(x) &= \frac{y_{i+1} - y_i}{h_i} - \frac{h_i}{6} (M_i(3t^2 - 6t + 2) + M_{i+1}(1 - 3t^2)), \\ K''(x) &= (1-t)M_i + tM_{i+1}, \\ K'''(x) &= \frac{-M_i + M_{i+1}}{h_i}. \end{aligned} \quad (6.19)$$

Interpoliranje druge derivacije u zadanim čvorovima samo po sebi ne osigurava i neprekidnost prve derivacije, te je taj uvjet potrebno konkretno zahtijevati. Analogno izrazu 6.9 za neprekidnost druge derivacije, možemo raspisati

$$\begin{aligned} K'(x_i + 0) &= \frac{y_{i+1} - y_i}{h_i} - h_i \frac{2M_i + M_{i+1}}{6}, \\ K'(x_i - 0) &= \frac{y_i - y_{i-1}}{h_{i-1}} - h_{i-1} \frac{2M_i + M_{i-1}}{6}, \end{aligned} \quad (6.20)$$

što uvjetuje neprekidnost prve derivacije polinom $K'(x)$. Za računanje kubičnog integracijskog polinoma (splinea) potrebno je znati derivacije u svakom integracijskom čvoru zato što nam vrijednost izraza derivacije, odnosno tangenta u točki interpolacijskog čvora, govori o trajektoriji funkcije. Kao takva nam omogućava bolje opisivanje kretne funkcije, odnosno preciznije rekonstruiranje krivulje originalne funkcije u odnosu na linearni spline (koji nema nikakve krivulje). Isto tako poznavanje derivacije funkcije u točki interpolacije omogućava bolju rekonstrukciju u odnosu na Newtonov ili Lagrangeov interpolacijski polinom, koji ne uzimaju u obzir neposrednu okolinu čvora na nikakav način, pa kao takvi samo povezuju čvorove interpolacije i pri tom često imaju velike oscilacije i fluktuacije.

7. Hermiteova interpolacija

Kao u prijašnjim primjerima različitih tipova interpolacije zahtjevanje da metoda interpolacije, odnosno tip interpolacijskog polinoma interpolira vrijednosti neke funkcije $y(x)$ u određenim čvorovima x_i , možemo zahtijevati i interpolaciju i derivacije funkcije $y'(x)$ u istim čvorovima x_i . Odnosno, tražimo polinom koji ispunjava uvjete

$$p(x_i) = y(x_i), p'(x_i) = y'(x_i), i = 0, 1, \dots, n.$$

Interpolacija koja ispunjava gore navedene uvjete zovemo Hermiteova interpolacija, a točke podataka x_i nazivamo Hermiteovi čvorovi. Ova metoda interpolacije korisna je ako imamo informacije o nagibu krivulje u određenim točkama, te želimo kvalitetnije kontrolirati te karakteristike u rekonstruiranoj funkciji. Izvedimo polinom za Hermiteovu interpolaciju preko uvjeta za interpolaciju i uvjeta za podudaranje u derivacijama. Ovaj polinom se u literaturi često izražava u obliku Lagrangeove interpolacijske formule ili Newtonove interpolacijske formule. U ovom izvodu koristiti ćemo Lagrangeov oblik. Pretpostavimo dva polinoma za koje vrijedi

$$p_{k,0}(x_i) = \begin{cases} 1, & i = k \\ 0, & i \neq k, \end{cases} \quad p'_{k,0}(x_i) = 0, i = 0, 1, \dots, n,$$

$$p'_{k,1}(x_i) = \begin{cases} 1, & i = k \\ 0, & i \neq k, \end{cases} \quad p_{k,1}(x_i) = 0, i = 0, 1, \dots, n.$$

Polinom $p_{k,0}$ koristiti ćemo za interpolaciju funkcije, a $p_{k,1}$ za interpolaciju njenih derivacija. Ovi polinomi su $2n + 1$ stupnja, te nakon što ih izračunamo preko njih možemo izraziti treći polinom

$$p_{2n+1}(x) = \sum_{k=0}^n (y_k p_{k,0}(x) + y'_k p_{k,1}(x)). \quad (7.1)$$

Derivacijom $\frac{dp_{2n+1}}{dx}$ dobivamo

$$p'_{2n+1}(x) = \sum_{k=0}^n (y_k p'_{k,0}(x) + y'_k p'_{k,1}(x)), \quad (7.2)$$

te iz toga možemo vidjeti ispunjavanje ranije navedenih uvjeta za interpolaciju i uvjeta za interpolaciju prve derivacije funkcije

$$p_{2n+1}(x) = \sum_{k=0}^n (y_k p_{k,0}(x) + y'_k p_{k,1}(x)) = y_k,$$

$$p'_{2n+1}(x) = \sum_{k=0}^n (y_k p'_{k,0}(x) + y'_k p'_{k,1}(x)) = y'_k.$$

Konstrukciju polinoma odraditi ćemo preko Lagrangeovog oblika interpolacijskog polinoma, odnosno recimo da su

$$p_{k,0}(x) = [1 - 2(x - x_k)L'_k(x_k)]L_k^2(x), \quad (7.3)$$

$$p_{k,1}(x) = (x - x_k)L_k^2(x), \quad (7.4)$$

gdje je $L_k(x)$ polinom Lagrangeove baze stupnja n . Polinome $p_{k,0}$ i $p_{k,1}$ se u literaturi često nazivaju funkcijama **Hermiteove baze** [1]. Poznavajući stupanj polinoma $L_k(x)$ i znajući da su polinomi $p_{k,0}$ i $p_{k,1}$ stupnja najviše $2n + 1$, možemo zaključiti da će i polinom p_{2n+1} biti najviše stupnja $2n + 1$.

7.1. Greška Hermiteove interpolacije

Pogreška Hermiteovog interpolacijskog polinoma $p_{2n+1}(x)$ je razlika vrijednosti u odnosu na funkciju y u čvorovima interpolacije $x_i, i = 0, 1, \dots, n$. Osnovni uvjeti Hermiteove interpolacije zahtijevaju jednakosti $y(x) = p_{2n+1}(x)$ i $y'(x) = p'_{2n+1}(x)$ stoga možemo pretpostaviti

$$y(x) - p_{2n+1}(x) \approx C\eta^2(x),$$

gdje je C neka konstanta, a $\eta(x)$ produkt svih članova $(x - x_i)$ na rasponu od 0 do n ($\eta(x) = \prod_{i=0}^n (x - x_i)$) Ako definiramo neku funkciju G za koju vrijedi

$$G(x) = y(x) - p_{2n+1}(x) - C\eta^2(x), \quad (7.5)$$

možemo primjetiti da funkcija G posjeduje nultočke u čvorovima interpolacije $x_i, i = 0, 1, \dots, n$. Odnosno, da je $G(x_i) = G'(x_i) = 0$. To je očekivano jer svojstvo interpolacije zahtijeva podudaranje originalne funkcije i rekonstruirane funkcije u svim točkama interpolacije, tj. čvorovima. No, ako odaberemo neku točku $x_{n+1} \in [a, b]$ koji ne pripada diskretnom skupu čvorova, onda je moguće definirati konstantu C tako da bude $G(x_{n+1}) = 0$. Iz toga znamo da funkcija $G(x)$ trenutno ima minimalno $n + 2$ nultočke, dok njena derivacija $G'(x)$ ima barem $n + 1$ nula raspoređene u točkama među njima. Isto tako ima nule u točkama interpolacije x_i , pa u konačnici ima minimalno $2n + 2$ nula. To znači da dvostruko derivirana funkcija $G''(x)$ ima jednu manje, odnosno minimalno $2n + 1$ nultočki, trostruko derivirana funkcija $G'''(x)$ ima $2n$ nultočki (itd. po

Rolleovom teoremu). U konačnici, derivirana funkcija $G^{(2n+2)}$ imati će barem jednu nultočku na promatramu rasponu $[a, b]$, te tu nultočku označavamo s ξ . Derivacijom funkcije $G(x)$ dobivamo izraz

$$G^{(2n+2)}\xi = y^{(2n+2)}\xi - C(2 + 2n)! = 0. \quad (7.6)$$

Iz te jednadžbe možemo izračunati konstantu C . Uvrštavanjem tog rezultata u jednadžbu greške dobivamo izraz

$$G(x_{n+1}) - p_{2n+1}(x) = \frac{y^{(2+2n)}(\xi)}{(2 + 2n)!} \eta^2(x_{n+1}), \quad (7.7)$$

gdje je $G(x_{n+1})$ ocijena greške funkcije, p_{2n+1} Hermiteov interpolacijski polinom, te η produkt svih članova $(x - x_i)$, $i = 0, 1, \dots, n$. Ova jednadžba je raspisana za proizvoljne točke $x_{i\pm 1}$, ali kako vidimo da s obje strane jednakosti mora biti nula, gore navedeni izraz za greške može se koristiti za bilo koje točke, uključujući i čvorove interpolacije x_i , $i = 0, 1, \dots, n$.

7.2. Newtonova forma Hermiteove interpolacije

Uz ranije izvedenu Lagrangeovu formu Hermiteove interpolacije (07.1), postoji i druga forma Hermiteove interpolacije zasnovana na Newtonovom obliku interpolacijskog polinoma. Ova forma koristi podijeljenje razlike s dvostrukim čvorovima [4]. Pretpostavimo dva čvora x_a i x_b , gdje je $x_b = x_a + k$, $k \rightarrow 0$. Onda je

$$f[x_a, x_b] = \lim_{k \rightarrow 0} [x_a, x_a + k] = \lim_{k \rightarrow 0} \frac{f(x_a + k) - f(x_a)}{k} = f'(x_a). \quad (7.8)$$

uz uvjet da je funkcija f derivabilna u x_a . Odnosno, vrijedi $f[x_a, x_b] = f[x_a, x_a] = f'(x_a)$.

Pretpostavimo dalje da imamo dvoje dvostrukih čvorova, čvor x_1 i njegov srodan čvor x_{1+k} , te čvor x_2 i njemu srodan čvor x_{2+k} , gdje je $k \rightarrow 0$. Podijeljene razlike za Newtonovu formu Hermiteove interpolacije za taj par dvostrukih čvorova računamo tablično na sljedeći način

\mathbf{x}	$f[x_i]$	$f[x_i, x_{i+1}]$	$f[x_i, x_{i+1}, x_{i+2}]$	$f[x_i, x_{i+1}, x_{i+2}, x_{i+3}]$
x_1	$f[x_1]$			
		$f'(x_1)$		
x_{1+k}	$f[x_1]$		$\frac{f[x_1, x_2] - f'(x_1)}{k}$	
		$f[x_1, x_2]$		$\frac{f'[x_1] + f'[x_2] - 2f[x_1, x_2]}{k^2}$
x_2	$f[x_2]$		$\frac{f'(x_2) - f[x_1, x_2]}{k}$	
		$f'(x_2)$		
x_{2+k}	$f[x_2]$			

Newtonov oblik Hermiteovog interpolacijskog polinoma je u tom slučaju jednak

$$\begin{aligned}
 p_{2n+1}(x) &= f[x_1] + f[x_1, x_1] \cdot (x - x_1) \\
 &\quad + f[x_1, x_1, x_2] \cdot (x - x_2)^2 \\
 &\quad + f[x_1, x_1, x_2, x_2] \cdot (x - x_1)^2 \cdot (x - x_2),
 \end{aligned} \tag{7.9}$$

uvažavajući da su

$$\begin{aligned}
 f[x_1, x_1] &= f'(x_1), \\
 f[x_1, x_1, x_2] &= \frac{f[x_1, x_2] - f[x_1, x_1+k]}{k} = \frac{f[x_1, x_2] - f'(x_1)}{k}, \\
 f[x_1, x_1, x_2, x_2] &= \frac{f[x_1, x_2, x_2] - f[x_1, x_1, x_2]}{x_2 - x_1} = \frac{\frac{f'(x_2) - f[x_1, x_2]}{k} - \frac{f[x_1, x_2] - f'(x_1)}{k}}{k} \\
 &= \frac{f'[x_1] + f'[x_2] - 2f[x_1, x_2]}{k^2}.
 \end{aligned}$$

Oba oblika Hermiteovog interpolacijskog polinoma koriste dodatne informacije o derivacijama interpolirane funkcije u čvorovima kako bi aproksimirali "glatkiju" funkciju. Oblici se razlikuju u načinu konstrukcije interpolacijskog polinoma. Lagrangeov oblik koristi istoimene Lagrangeove interpolacijske polinome, detaljnije obrađene u poglavlju 4. Newtonov oblik Hermiteovog interpolacijskog polinoma koristi podijeljene razlike. U konačnici, odabir između ova dva oblika interpolacijskih polinoma ovisi o složenosti problema i zahtjevima implementacije podataka. Nakon računanja, obje metode konstruiraju valjale polinome, ali se mogu razlikovati u smislu stabilnosti i numeričke preciznosti ovisno o promatranom problemu.

Primjer 7.1. *Pokusom simulacije procesa zaleta asinkronog motora dobivene su vrijednosti mehaničkog momenta na osovini stroja u ovisnosti o kutnoj brzini rotora u zadanim trenucima tijekom zaleta, prikazane u tablici. Izmjerene su vrijednosti momenta pri 3 različite brzine vrtnje.*

M [Nm]	ω [rad/s]	$\frac{dM}{d\omega}$
20	0	3
35	5	5
30	10	-10

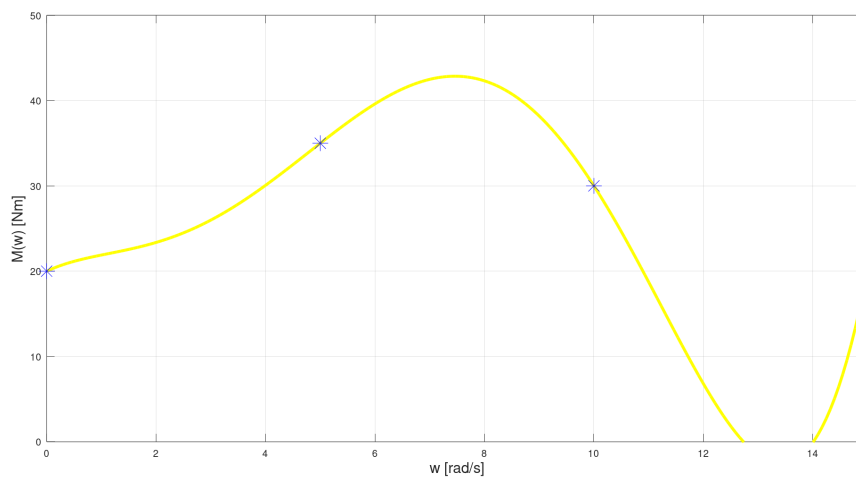
Tablica 7.1. Izmjerene vrijednosti momenta i brzine

Karakteristike koje opisuju moment u ovisnosti o brzini vrtnje u teoriji elektromotornih pogona nazivaju se izlazne karakteristike. Karakteristične točke za izlazne karakteristike asinkronog stroja su potezni moment (moment pri brzini 0), prekretni moment (maksimalna vrijednost momenta), te brzina praznog hoda (brzina pri momentu jednakom $M(\omega) = 0$). Vrijednost poteznog momenta imamo kao zadanu ($M(0) = 20\text{Nm}$), druge dvije možemo odrediti nakog računanja polinoma.

Uvrštavanjem izmjerenih vrijednosti u izraz za Newtonovu formu Hermiteovog interpolacijskog polinoma dobivamo

$$\begin{aligned}
 M(\omega) &= 20 + 3\omega - \frac{17}{10}\omega^2 + \frac{67}{100}\omega^3 - \frac{2}{25}\omega^4 + \frac{7}{2500}\omega^5 \\
 &= 20 + 3\omega - 1.7\omega^2 + 0.67\omega^3 - 0.08\omega^4 + 0.0028\omega^5.
 \end{aligned}
 \tag{7.10}$$

Grafički prikaz izračunatog polinoma možemo vidjeti na slici 7.1.



Slika 7.1. Polinom $M(\omega)$ izračunat preko Hermiteovog interpolacijskog polinoma prikazan preko čvorova interpolacije

Sa slike je vidljivo da je polinom povezao sve točke interpolacije glatkom funkcijom $M(\omega)$. Uvrštavanjem vrijednosti kutne brzine u izračunati polinom možemo dobiti iznos mehaničkog momenta u bilo kojoj točki karakteristike, što nam olakšava predviđanje ponašanja asinkronog motora u područjima van skupa interpolacijskih čvorova.

8. Gaussove formule

Gaussove formule su težinske integracijske formule. Gaussova integracija omogućava precizne rezultate pri numeričkom računanju integrala, a time i olakšava proces rješavanja problema koji se inače ne bi mogli analitički izračunati.

Gaussove integracijske formule zapisuju se u obliku

$$\int_a^b \omega(x)y(x)dx \approx \sum_{k=1}^n \omega_k y(x_k), \quad (8.1)$$

gdje je ω težinska funkcija koja je pozitivna na rasponu (a, b) , $y(x)$ funkcija čije vrijednosti interpoliramo, a ω_k su težinski koeficijenti, u literaturi nazivane i **težine** integracijskih formula. Težine integracijskih formula su matematički faktori koji se koriste za određivanje težinske vrijednosti određenih čvorova pri izračunu numeričkog integrala. Svaki interpolacijski čvor ima svoj specifični težinski koeficijent koji se množi s vrijednošću interpolacijske funkcije. Težinski koeficijenti izabiru se kako bi minimizirali pogrešku interpolacije i osigurali točnost riješenog integrala. Određeni su na temelju poznatih ili odabranih čvorova, najčešće pomoću metoda ortogonalnosti polinoma.

Postoje posebni slučajevi težinskih funkcija koje se integriraju na specifičnim intervalima, te za njih postoje standardne težine i čvorovi koje možemo naći u literaturi [2]. Neki od primjera su

ω	interval	Gauss-
1	$[-1, 1]$	Legendreove
e^{-x^2}	$(-\infty, \infty)$	Hermiteove
$\frac{1}{\sqrt{1-x^2}}$	$[-1, 1]$	Čebiševljeve

Ako želimo da formula za integraciju točno integrira polinome što je većeg stupnja moguće, tada gledamo čvorove integracije x_k kao nultočke polinoma koji su ortogonalni na intervalu (a, b) u odnosu na težinsku funkciju ω . Težinske koeficijente eksplicitno računamo po formuli

$$\omega_k = \int_a^b \omega(x)L_k(x)dx, \quad k = 1, \dots, n, \quad (8.2)$$

gdje je $L_k(x)$ polinom Lagrangeove baze stupnja $n - 1$ kojeg smo koristili i pri računanju Hermiteovih baza interpolacijskog polinoma (7.3) i (7.4). Uzimajući to u obzir, bitno je napomenuti

da se formule za Gaussovu integraciju mogu izvesti integriranjem Hermiteovog interpolacijskog polinoma [1]. Raspišimo ponovno Lagrangeov oblik Hermiteovog interpolacijskog polinoma,

$$\begin{aligned} p_{2n+1}(x) &= \sum_{k=0}^n (y_k p_{k,0}(x) + y'_k p_{k,1}(x)) \\ &= \sum_{k=1}^n ((1 - 2(x - x_k)L'_k(x_k))L_k^2(x)y_k + (x - x_k)L_k^2(x)y'_k). \end{aligned}$$

Integriranjem tog izraza dobivamo

$$\int_a^b \omega(x) p_{2n-1}(x) dx = \sum_{k=1}^n (\alpha_k y_k + \beta_k y'_k), \quad (8.3)$$

gdje su α i β koeficijenti

$$\begin{aligned} \alpha_k &= \int_a^b \omega(x) [1 - 2(x - x_k)L'_k(x_k)] L_k^2(x) dx, \\ \beta_k &= \int_a^b \omega(x) (x - x_k) L_k^2(x) dx. \end{aligned} \quad (8.4)$$

Bitno je spomenuti da odabir ispravnih težinskih koeficijenata, odnosno težina, znatno utječe na točnost Gaussove integracijske formule. Zato se korektan izbor težinskih koeficijenata provodi uzimajući u obzir konkretna i individualna matematička svojstva promatranog problema, i kad je to moguće, pažljivim odabirom čvorova.

9. Zaključak

U ovom radu obrađuje se pojam aproksimacije i grešaka pri aproksimaciji, te više različitih načina interpoliranja. Ustanovljeno je da se aproksimacija javlja u dva oblika. Prvi, kad nam je poznata funkcija, ali želimo je zamjeniti nekom drugom, recimo jednostavnijom ili povoljnijom za računanje. Drugi, kad su nam poznate samo neke vrijednosti funkcije, te pokušavamo "rekonstruirati" funkciju, odnosno pronaći funkciju koja najbolje opisuje sve poznate vrijednosti te omogućava pregled vrijednosti između ili izvan poznatih. Ovaj oblik je češći u praksi te je u ovom radu objašnjen u nekoliko primjera kroz interpolaciju. Lagrangeov oblik interpolacijskog polinoma je koristan u teoriji. Pogotovo u slučajevima kad moramo više puta interpolirati iste vrijednosti ili aproksimirati funkciju u slučaju gdje znamo koliko je vrijednosti potrebno za adekvatnu aproksimaciju. S druge strane, ako otkrivamo, odnosno dodajemo nove vrijednosti, polinom moramo svaki put nanovo računati, što povećava potrebno vrijeme i nepotrebno troši računalnu memoriju. S druge strane, Newtonov oblik interpolacijskog polinoma omogućuje znatno brže dodavanje novootkrivenih vrijednosti. Stoga je bolji u slučajevima gdje npr. ne znamo koliko je vrijednosti potrebno da se neka zadana funkcija aproksimira sa zadovoljavajućom točnošću, odnosno kada moramo stupnjevito povećavati broj vrijednosti, to jest čvorova interpolacije.

Spline interpolacije obrađene su u dva osnovna oblika, linearna spline interpolacija i kubična spline interpolacija. Spline interpolacija razdvaja funkciju na podintervale ograničene jednim parom čvorova interpolacije, te opisuje funkciju zasebno na svakom podintervalu. Linearna spline interpolacija to čini linearnim pravcima, dok kubična spline interpolacija to čini polinomima trećeg stupnja, odnosno kubičnim polinomima. Kod kubičnih polinoma bitno je poznavati i vrijednost derivacija funkcije u svakom čvoru koja nam omogućava precizniju rekonstrukciju originalne funkcije bez nekarakterističnih oscilacija i fluktuacija interpolacijske funkcije. Spline interpolacije su iznimno korisne u konstruiranju karakteristika koje opisuju promjene fizikalnih veličina u međusobnom odnosu, jer svojom konstrukcijom "prigušuju" nekarakteristične oscilacije koje bi inače neadekvatno opisale odnose npr. okrente brzine rotora i mehaničkog momenta električnog stroja.

Hermiteova interpolacija koristi ne samo funkcijske vrijednosti, već i vrijednosti derivacija funkcije u interpolacijskim čvorovima. Ovaj oblik interpolacije je koristan pri rekonstrukciji nelinearnih signala. Proučene su dvije forme Hermiteovog interpolacijskog polinoma - Lagrangeova forma i Newtonova forma. Lagrangeova forma Hermiteovog interpolacijskog polinoma koristi Lagrangeov polinom, koji se dobiva direktno iz definicije i uvjeta polinomne interpolacije. Newtonova forma Hermiteovog interpolacijskog polinoma temelji se na Newtonovim diferencama, odnosno na podijeljenim razlikama. Ovaj oblik omogućuje učinkovitije računanje interpolacijskog polinoma, te je pogodniji za računalnu implementaciju i numeričke proračune. Iako je Newtonova forma efikasnija za računalnu implementaciju, Lagrangeova forma ima prednosti u tome što je intuitivnija za razumijevanje. Stoga, konačan izbor između ove dvije forme ovisiti će o tipu primjene

i preferenciji inženjera.

Gaussove formule omogućavaju efikasno numeričko računanje integrala. Ove formule se često primjenjuju u poljima poput računalnih simulacija ili elektroničke analize. Jedna od primjena Gaussovih formula je u analizi elektromagnetskih polja. Elektromagnetska polja imaju prostorno nejednaku raspodjelu naboja i gustoću, te koristeći Gaussove formule numerički se mogu izračunati te vrijednosti. Slična primjena je moguća pri projektiranju kondenzatora, odnosno kapacitivnih elemenata gdje se integralom može opisati raspodjela naboja po kondenzatorskim pločicama. U konačnici, Gaussove formule omogućuju efikasno rješavanje kompleksnih problema u elektrotehnici, i kao takve olakšavaju projektiranje, analizu i optimizaciju električnih mreža i sustava.

Bibliografija

- [1] Drmač, Z.; Hari V.; Marušič, M.; Rogina, M.; Singer, S.; Singer, S.: "Numerička analiza", Sveučilište u Zagrebu, PMF - Matematički odjel, Zagreb, 2003.
- [2] Kaw, Autar: Direct Method of Interpolation - More examples in electrical engineering, University of South Florida, Tampa, 2009.
- [3] Scitovski, Rudolf: "Numerička matematika: Drugo izdanje", Sveučilište J. J. Strossmayera u Osijeku, Odjel za matematiku, Osijek, 2004.
- [4] Singer, Saša: Numerička matematika, predavanje 6, PMF-Matematički odsjek, Zagreb, 2010. (01.07.2023.)
- [5] Bosner, Nela: Interpolacija i aproksimacija splajnovima, predavanje 5, PMF-Matematički odsjek, Zagreb. (01.07.2023.)

Sažetak i ključne riječi

U ovom završnom radu opisuje se polinomna interpolacija i primjene različitih metoda interpolacije u elektrotehnici. Obrazložuju se greške te navode tipovi grešaka koji su česti pri interpolaciji i aproksimaciji. Proučavaju se različite metode interpolacije, odnosno oblici interpolacijskih polinoma - Lagrangeov, Newtonov, Hermiteov, te spline interpolacija. Prikazuju se izvodi formula za pojedinačne oblike interpolacijskih polinoma, te se kroz praktične primjere opisuju i uspoređuju u kontekstu primjene u različitim granama elektrotehnike.

Ključne riječi: Interpolacija, aproksimacija, greške, Lagrangeov oblik interpolacijskog polinoma, Newtonov oblik interpolacijskog polinoma, Spline interpolacija, Hermiteova interpolacija, Gaussove formule.

Summary and key words

This paper describes polynomial interpolation, that is, the calculation of values between two or more known points or values, as well as describing the calculated values using polynomial functions. It explains errors and describes the types of errors which are common with interpolation. It examines different types of interpolation polynomials - Lagrange, Newtons, spline and Hermites. It shows the derivation of formulas for those types of interpolation polynomials and describes them through examples, and compares them in the context of application within electrical engineering.

Keywords: Interpolation, approximation, errors, polynomials, Lagrange, Newton, spline interpolation, Hermite, Gauss.

A Lista Octave kodova korištenih za primjere

Primjer 4.1

```
1 pkg load symbolic;
2
3 syms x y p p1 p2 p3 p4 p5;
4 xn = [1 3 4 7 10];
5 yn = [2 5 1 6 10];
6 x = linspace(0,12,50)
7 p1 = (((x-3).*(x-4).*(x-7).*(x-10)).*2)/((1-3).*(1-4).*(1-7).*(1-10));
8 p2 = (((x-1).*(x-4).*(x-7).*(x-10)).*5)/((3-1).*(3-4).*(3-7).*(3-10));
9 p3 = (((x-1).*(x-3).*(x-7).*(x-10)).*1)/((4-1).*(4-3).*(4-7).*(4-10));
10 p4 = (((x-1).*(x-3).*(x-4).*(x-10)).*6)/((7-1).*(7-3).*(7-4).*(7-10));
11 p5 = (((x-1).*(x-3).*(x-4).*(x-7)).*10)/((10-3).*(10-4).*(10-7).*(10-1));
12 p = p1+p2+p3+p4+p5;
13
14
15
16 plot(x,p, 'y-', 'linewidth', 3, xn, yn, '*b', 'markersize', 16);
17 axis([0 12 0 12])
18 grid on;
19 xlabel("x", 'fontsize', 15);
20 ylabel("Pn(x)", 'fontsize', 15);
21
```

Slika A1. Octave kod za primjer 4.1

Primjer 4.2

```
1 x = linspace(-10,10,100)
2 y = sin(x)
3 #plot(x,y)
4 axis([-10 10 -2 2])
5 grid on
6 xlabel("X", 'fontsize', 12)
7 ylabel("sin(x)", 'fontsize', 12)
8
9 sym p;
10 x1 = [-10 -5 0 5 10]
11 y1 = [0.54402 0.9589 0 -0.95892 -0.54402]
12
13 #plot(x1,y1, '*b', 'markersize',16)
14 axis([-10 10 -2 2])
15 grid on
16 xlabel("X", 'fontsize', 12)
17 ylabel("y", 'fontsize', 12)
18
19 p1 = (((x+5).*(x-0).*(x-5).*(x-10)).*0.54402)/((-10+5).*(-10-0).*(-10-5).*(-10-10));
20 p2 = (((x+10).*(x-0).*(x-5).*(x-10)).*0.9589)/((-5+10).*(-5-0).*(-5-5).*(-5-10));
21 p3 = (((x+5).*(x-0).*(x-5).*(x-10)).*0)/((-10+5).*(-10-0).*(-10-5).*(-10-10));
22 p4 = (((x+5).*(x-0).*(x+10).*(x-10)).*(-0.95892))/((5+10).*(5+5).*(5+0).*(5-10));
23 p5 = (((x+5).*(x-0).*(x-5).*(x+10)).*(-0.54402))/((10+5).*(10+10).*(10-5).*(10-0));
24
25 p = p1+p2+p3+p4+p5;
26
27 plot(x,p, 'y-', 'linewidth', 3, x1, y1, '*b', 'markersize', 16);
28 axis([-10 10 -2 2]);
29 grid on;
30 xlabel("x", 'fontsize', 15);
31 ylabel("Pn(x)", 'fontsize', 15);
```

Slika A2. Octave kod za prvi dio primjera 4.2

```

1 sym p;
2 x1 = [-10 -8 -6 -4 -2 0 2 4 6 8 10];
3 y1 = [0.54402 -0.989 0.279 0.757 -0.909 0 0.909 -0.757 -0.279 0.989 -0.54402];
4
5 x = linspace(-12,12,100);
6
7 p1 = (((x+8).*(x+6).*(x+4).*(x+2).*(x-0).*(x-2).*(x-4).*(x-6).*(x-8).*(x-10)).*(0.54402))/((-10+8).*(-10+6).*(-10+4).*(-10+2).*(-10+0).*(-10-2).*(-10-4).*(-10-6).*(-10-8).*(-10-10));
8
9 p2 = (((x+10).*(x+6).*(x+4).*(x+2).*(x-0).*(x-2).*(x-4).*(x-6).*(x-8).*(x-10)).*(-0.989))/((-8+10).*(-8+6).*(-8+4).*(-8+2).*(-8+0).*(-8-2).*(-8-4).*(-8-6).*(-8-8).*(-8-10));
10
11 p3 = (((x+10).*(x+8).*(x+4).*(x+2).*(x-0).*(x-2).*(x-4).*(x-6).*(x-8).*(x-10)).*(0.279))/((-6+10).*(-6+8).*(-6+4).*(-6+2).*(-6+0).*(-6-2).*(-6-4).*(-6-6).*(-6-8).*(-6-10));
12
13 p4 = (((x+10).*(x+8).*(x+6).*(x+2).*(x-0).*(x-2).*(x-4).*(x-6).*(x-8).*(x-10)).*(0.757))/((-4+10).*(-4+6).*(-4+8).*(-4+2).*(-4+0).*(-4-2).*(-4-4).*(-4-6).*(-4-8).*(-4-10));
14
15 p5 = (((x+10).*(x+8).*(x+6).*(x+4).*(x-0).*(x-2).*(x-4).*(x-6).*(x-8).*(x-10)).*(-0.909))/((-2+10).*(-2+6).*(-2+4).*(-2+2).*(-2+0).*(-2-2).*(-2-4).*(-2-6).*(-2-8).*(-2-10));
16
17 p7 = (((x+10).*(x+8).*(x+6).*(x+4).*(x+2).*(x-0).*(x-2).*(x-4).*(x-6).*(x-8).*(x-10)).*(0.909))/((2+10).*(2+8).*(2+6).*(2+4).*(2+2).*(2+0).*(2-2).*(2-6).*(2-8).*(2-10));
18
19 p8 = (((x+10).*(x+8).*(x+6).*(x+4).*(x+2).*(x-0).*(x-2).*(x-4).*(x-6).*(x-8).*(x-10)).*(-0.757))/((4+10).*(4+8).*(4+6).*(4+4).*(4+2).*(4+0).*(4-2).*(4-6).*(4-8).*(4-10));
20
21 p9 = (((x+10).*(x+8).*(x+6).*(x+4).*(x+2).*(x-0).*(x-2).*(x-4).*(x-6).*(x-8).*(x-10)).*(-0.279))/((6+10).*(6+8).*(6+6).*(6+4).*(6+2).*(6+0).*(6-2).*(6-4).*(6-6).*(6-8).*(6-10));
22
23 p10 = (((x+10).*(x+8).*(x+6).*(x+4).*(x+2).*(x-0).*(x-2).*(x-4).*(x-6).*(x-8).*(x-10)).*(0.989))/((8+10).*(8+8).*(8+6).*(8+4).*(8+2).*(8+0).*(8-2).*(8-4).*(8-6).*(8-8).*(8-10));
24
25 p11 = (((x+10).*(x+8).*(x+6).*(x+4).*(x+2).*(x-0).*(x-2).*(x-4).*(x-6).*(x-8).*(x-10)).*(-0.54402))/((10+10).*(10+8).*(10+6).*(10+4).*(10+2).*(10+0).*(10-2).*(10-4).*(10-6).*(10-8));
26
27 p = p1+p2+p3+p4+p5+p7+p8+p9+p10+p11;
28
29 plot(x,p, 'y-', 'linewidth', 3, x1, y1, 'b', 'markersize', 16);
30 axis([-10 10 -2 2]);
31 grid on;
32 xlabel("x", 'fontsize', 18);
33 ylabel("Pn(x)", 'fontsize', 15);

```

Slika A3. Octave kod za drugi dio primjera 4.2

Primjer 5.1

```

1 pkg load symbolic;
2
3 syms x y p p1 p2 p3 p4 p5;
4 xn = [1 3 5 7];
5 yn = [2 7 3 4];
6 x = linspace(0,10,50)
7
8 p = 2+ (x-1).*(5/2)+(x-1).*(x-3).*(-9/8)+(x-1).*(x-3).*(x-7).*(7/24)
9
10 plot(x,p, 'y-', 'linewidth', 3, xn, yn, 'b', 'markersize', 16,);
11 axis([0 10 0 10])
12 grid on;
13 xlabel("x", 'fontsize', 15);
14 ylabel("Pn(x)", 'fontsize', 15);
15

```

Slika A4. Octave kod za prvi dio primjera 5.1

```

1 pkg load symbolic;
2
3 syms x y p p1 p2 p3 p4 p5;
4 xn = [1 3 5 7 10 12];
5 yn = [2 7 3 4 5 10];
6 x = linspace(0,15,50)
7
8 p = (17/3080)*(x.*x.*x.*x.*x) - (41/220)*(x.*x.*x.*x) + (21577/9240)*(x.*x.*x)-(20257/1540)*(x.*x) + (72869/2310)*(x) - 204/11
9
10 plot(x,p, 'y-', 'linewidth', 3, xn, yn, 'b', 'markersize', 16,);
11 axis([0 15 0 15])
12 grid on;
13 xlabel("x", 'fontsize', 15);
14 ylabel("Pn(x)", 'fontsize', 15);
15

```

Slika A5. Octave kod za drugi dio primjera 5.1