Aproksimacija i iterpolacija funkcije

Mesec, Bruno

SVEUČILIŠTE U ZAGREBU

FAKULTET ORGANIZACIJE I INFORMATIKE

V A R A Ž D I N

Bruno Mesec

APROKSIMACIJA I INTERPOLACIJA FUNKCIJE

ZAVRŠNI RAD

Varaždin, 2018.

SVEUČILIŠTE U ZAGREBU

FAKULTET ORGANIZACIJE I INFORMATIKE

V A R A Ž D I N

Bruno Mesec

Matični broj: 43400/14–R

Studij: Informacijski sustavi

APROKSIMACIJA I INTERPOLACIJA FUNKCIJE

ZAVRŠNI RAD

Mentorica:

Dipl.inž. Marija Jakuš

Varaždin, rujan 2018.

Bruno Mesec

Izjava o izvornosti

Izjavljujem da je moj završni rad izvorni rezultat mojeg rada te da se u izradi istoga nisam koristio drugim izvorima osim onima koji su u njemu navedeni. Za izradu rada su korištene etički prikladne i prihvatljive metode i tehnike rada.

Autor potvrdio prihvaćanjem odredbi u sustavu FOI-radovi

Sažetak

Ovaj rad sastoji se od dvije glavne komponente, a one su aproksimacija funkcije i interpolacija funkcije. Aproksimaciju funkcije možemo ukratko opisati kao slučaj kada na nekom skupu zamijenimo zadanu funkciju nekom drugom. Interpolaciju možemo definirati kao zahtjev da se vrijednosti nekih dviju funkcija podudaraju na određenom skupu argumenata, to jest točaka. Osim ovih dviju glavnih komponenti, rad će sadržavati i obradu Taylorovog polinoma, Lagrangeovog interpolacijskog polinoma za ekvidistantne i neekvidistantne čvorove, Newtonovog oblika interpolacijskog polinoma za ekvidistantne i neekvidistantne čvorove, kao i obradu Hermiteove interpolacije, numeričkog deriviranja, Čebiševljevih polinoma prve i druge vrste, linearnih i B-spline-ova i ocjenu pogreške za svaku od metoda. Također, bit će implementiran algoritam u nekom programskom jeziku za neku metodu aproksimacije/interpolacije.

Ključne riječi: matematika, aproksimacija, interpolacija, funkcija, polinom, greška, spline

Sadržaj

1. Uvod

U matematici je razvijeno mnogo različitih postupaka aproksimativnog, to jest približnog računanja matematičkih problema. Takve postupke nazivamo numeričkim postupcima, a odgovarajuće područje matematike, koje se bavi takvim postupcima, zove se numerička matematika. Numerička matematika je, kao i sve ostale grane matematike, vrlo opsežna i u stalnom je razvoju.

Postoji mnogo različitih vrsta algebarskih funkcija koje se mogu koristiti kao približne funkcije, ali najčešće korištene funkcije za tu svrhu su polinomi, eksponencijalne funkcije, trigonometrijske funkcije, a u novije vrijeme i racionalne funkcije. Koja god vrsta funkcija se koristila, mora imati sljedeće značajke: lako određivanje i izračunavanje, a isto tako i lako diferenciranje i integriranje.

Ovaj završni rad obuhvaća dva različita pristupa kojim se određuju približne funkcije koje se koriste za opisivanje zavisnosti skupa podataka, a to su aproksimacija koju možemo opisati kao približno poklapanje, i interpolacija, to jest točno poklapanje.

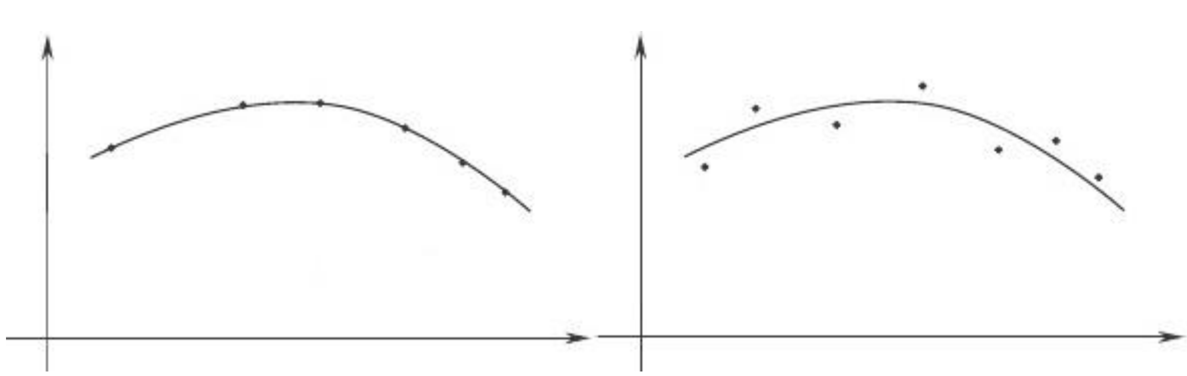
Aproksimacija funkcije je matematički pojam koji označava postupak pronalaženja funkcije koja približno aproksimira, odnosno opisuje neki konačni skup točaka ili neku drugu funkciju. Aproksimaciju funkcije možemo opisati i na način da zadanu funkciju zamijenimo nekom drugom funkcijom, koja nije ista kao originalna, ali je dovoljno točna i lakše ju je izračunati. Problem aproksimacije često se pojavljuje u teoriji, ali još češće u praksi. Potreba za aproksimacijom javila se kao posljedica činjenice da ne možemo uvijek pronaći točno rješenje nekog matematičkog problema pa moramo potražiti najbolju moguću aproksimaciju tog rješenja. Aproksimacija može biti linearna i nelinearna po parametrima. Javlja se u dva oblika, tako da nam je originalna funkcija f poznata, ali ima formu koja je prekomplicirana za računanje, ili da nam je funkcija f nepoznata, ali su nam poznate neke informacije o njoj, npr. poznate su nam vrijednosti te funkcije na određenom skupu točaka. U prvom slučaju biramo informacije o funkciji koje ćemo koristiti za aproksimaciju i kod tog slučaja se može ocijeniti greška dobivene aproksimacije, a u drugom slučaju se ne može napraviti ocjena greške bez dodatnih informacija o nepoznatoj funkciji f. [1], [2], [3]

Interpolaciju u numeričkoj matematici možemo opisati kao zahtjev da se vrijednosti nekih dviju funkcija podudaraju na određenom skupu argumenata, to jest točaka, koje nazivamo čvorovima interpolacije. U složenijoj problematici može se dodati zahtjev za podudaranjem i u nekim derivacijama. Isto tako, interpolacijom se smatra i svako izračunavanje nove točke između već postojećih točaka podataka. Osnovna zadaća interpolacije je pogodnu krivulju provući kroz niz točaka $(x_i, y_i), i = 0,1, \dots, n$. Numerički to postižemo pomoću funkcije $\varphi(x)$ koja u točkama x_i, tzv. čvorovima, poprima zadane vrijednosti y_i, to jest $\varphi(x)$ zadovoljava uvjet interpolacije $\varphi(x_i) = y_i,\ i = 0,1, \dots, n$. [1], [2], [3]

Postupak interpolacije može se napraviti na dva načina. Prvi način je da se najprije odredi interpolacijska funkcija prema zadanim ulaznim točkama, a nakon toga se izračuna interpolacijska funkcija u nekoj željenoj točki x. Međutim, ovaj način nije baš dobro rješenje u praksi zato što je neefikasan i više podložan greški. Bolje je upotrijebiti metodu kojom se računanje interpolacije započinje u ulaznoj točki koja je najbliža traženoj točki x, a zatim se zbrajaju sve manje i manje korekcijske vrijednosti sve do kad se ne dođe do tražene vrijednosti funkcije u zadanoj točki. Zadnja pridodana vrijednost je najmanja i ona predstavlja ocjenu greške. Red interpolacije određuje broj ulaznih točaka, i on je jednak tom broju umanjenom za 1. [4]

Za interpolaciju funkcija najčešće koristimo interpolacijske polinome od kojih se ističu Lagrangeov oblik interpolacijskog polinoma i Newtonov oblik interpolacijskog polinoma. Lagrangeov oblik se najčešće koristi u teoretske svrhe, dok se Newtonov oblik obično koristi u praksi. Također, u praksi se ne koriste interpolacijski polinomi stupnja većeg od tri, jer greške nastale prilikom zaokruživanja koje radi računalo mogu biti vrlo velike. U tim slučajevima najbolje je koristiti se spline-ovima ili metodom najmanjih kvadrata.

Interpolacijom dolazimo do funkcija koje točno prolaze kroz sve zadane točke, a aproksimacijom do funkcija koje prolaze kroz određenu grupu podataka na najbolji mogući način, međutim, ne moraju točno prolaziti kroz sve zadane točke. To možemo vidjeti na sljedećoj slici. [3]



Slika 1: Interpolacija (lijevo) i aproksimacija (desno)
Izvor: [*http://www.grad.hr/nastava/matematika/mat3/node128.html*]

Kada određujemo približnu funkciju, pri tome radimo neku grešku, jer se dobivena funkcija ne poklapa u svim čvorovima s originalnom funkcijom. Razlika između vrijednosti približne funkcije i originalne funkcije predstavlja grešku. Uzroka takvih grešaka ima više. Često postoji nedostatak u povezanosti određenog matematičkog problema i stvarnog problema kojeg taj matematički problem opisuje. Zatim dolaze greške u početnim podacima koje su uzrokovane pogrešnim mjerenjima tih podataka. Treću grupu uzroka, koja je za nas najvažnija i koju ćemo ovdje obrađivati, čine greške koje se pojavljuju u metodama s kojima rješavamo određeni matematički problem. Na kraju, tu su greške zaokruživanja koje se javljaju prilikom izvođenja aritmetičkih operacija. Kao što smo rekli, u ovom ćemo se radu posvetiti trećoj grupi pogrešaka pa ćemo za svaku metodu odrediti ocjenu greške. [5]

2. Opći problem aproksimacije

U uvodu smo rekli ponešto o samoj aproksimaciji, a u ovom ćemo poglavlju malo detaljnije objasniti što je to problem aproksimacije. Ako su nam poznate određene informacije o funkciji f, definiranoj na podskupu $X \subseteq \mathbb{R}$, na osnovu tih informacija želimo funkciju f zamijeniti drugom funkcijom φ, koja se nalazi u istom skupu X ili na još većem skupu, tako da su f i φ bliske u određenom smislu. Skup X najčešće je diskretni skup točaka ili interval oblika $[a, b]$. [6]

Kao što smo već objasnili, problem aproksimacije javlja se u dva različita oblika: [7]

i) Znamo funkciju f analitički ili slično, ali je njezin oblik vrlo kompliciran pa je računanje vrijednosti funkcije previše složen zadatak. U tom se slučaju odabiru informacije o funkciji f koje će se koristiti i po nekom kriteriju se odredi aproksimacijska, to jest približna funkcija φ. Prednosti ovog oblika su što mi sami biramo informacije o funkciji f koje ćemo koristiti te kod ovog oblika možemo ocijeniti grešku aproksimativne funkcije φ.

ii) Kod ovog problema aproksimacije ne znamo formu funkcije f, ali znamo neke informacije o njoj, na primjer vrijednosti funkcije na nekom skupu točaka. Približna funkcija φ tada se određuje pomoću informacija o funkciji f koje su nam dostupne. Te informacije, osim samih podataka, mogu uključivati i očekivani oblik ponašanja tih podataka, odnosno oblik zamjenske funkcije φ. Kod ovog oblika problema aproksimacije, ne možemo ocijeniti grešku aproksimativne funkcije bez dodatnih informacija o originalnoj funkciji f.

Što se tiče primjene tih problema aproksimacije, prvi oblik više se pojavljuje u teoriji, za razvoj određenih metoda na bazi aproksimacije, kao što su rješavanje diferencijalnih jednadžbi ili integriranje funkcija, a drugi oblik više se pojavljuje u praksi, npr. kod mjerenja nekih veličina. [7]

Kad funkciju φ zapišemo u obliku

$$\varphi(x) = \varphi(x; a_0, a_1, \dots, a_m),$$

odnosno kao funkciju koja ovisi i o parametrima a_k, onda kažemo da smo izabrali opći oblik aproksimacijske funkcije φ. Oblici aproksimacije funkcije mogu se podijeliti na: [7]

i) Linearne aproksimacijske funkcije,

ii) Nelinearne aproksimacijske funkcije.

2.1. Linearne aproksimacijske funkcije

Opći oblik linearnih aproksimacijskih funkcija je

$$\varphi(x) = a_0\varphi_0(x) + a_1\varphi_1(x) + \cdots + a_m\varphi_m(x)$$

gdje su $\varphi_0, \ldots, \varphi_m$ funkcije koje su nam poznate i koje znamo izračunati. Linearnost se odnosi na ovisnost aproksimativne funkcije φ o njenim parametrima a_k koji se trebaju odrediti. Oni su, dakle, koeficijenti u linearnoj kombinaciji poznatih funkcija. Prednost ovog oblika aproksimacije je ta da određivanje parametara a_k obično vodi na sustave linearnih jednadžbi ili linearne probleme optimizacije, što je dobro jer njih možemo lakše riješiti nego nelinearne probleme. [7]

Oblici linearnih aproksimacijskih funkcija koji se najčešće koriste su: [6]

1. Algebarski polinomi, $\varphi_k(x) = x^k,\ k = 0, \ldots, m$, to jest
$$\varphi(x) = a_0 + a_1x + \cdots + a_mx^m.$$
2. Trigonometrijski polinomi, koji su pogodni za aproksimaciju periodičkih funkcija. Za funkciju φ_k uzima se $(m+1)$-na funkcija iz skupa $\{1, \cos x, \sin x, \cos 2x, \sin 2x, \ldots\}$, gdje je $k = 0, \ldots, m$.
3. Po dijelovima polinomi, tzv. spline funkcije koje se na svakom podintervalu svode na polinom niskog stupnja. Ti polinomi su na svim podintervalima istog stupnja, to jest $\varphi|_{[x_{k-1},x_k]} = p_k,\ k = 1, 2, \ldots, n$, gdje su $x_0, \ldots, x_n$ zadane točke.

2.2. Nelinearne aproksimacijske funkcije

Nelinearne aproksimacijske funkcije zapisujemo u obliku

$$\varphi(x) = \varphi(x; a_0, a_1, \ldots, a_m).$$

One imaju nelinearnu ovisnost o parametrima aproksimacijske funkcije $a_0, \ldots, a_m$. Određivanje parametara a_k obično vodi na sustave nelinearnih jednadžbi ili nelinearne probleme optimizacije. [7]

Oblici nelinearnih aproksimacijskih funkcija koje se najčešće koriste su: [6]

1. Eksponencijalne funkcije $\varphi(x) = c_0e^{b_0x} + c_1e^{b_1x} + \cdots + c_re^{b_rx}$, koje imaju $n = 2r + 2$ nezavisna parametra. Te funkcije imaju primjenu u raznim granama, kao što su biologija, ekonomija, medicina, a mogu opisivati procese rasta i odumiranja kod populacije.

2. Racionalne funkcije $\varphi(x) = \frac{b_0+b_1x+\cdots+b_rx^r}{c_0+c_1x+\cdots+c_sx^s}$, koje imaju bolja svojstva aproksimacije nego polinomi. Kod tog oblika aproksimacije broj nezavisnih parametara je $n = r + s + 1$.

2.3. Kriteriji aproksimacije

Aproksimacijske funkcije biraju se tako da na najbolji mogući način zadovolje uvjete koji se postavljaju pred njih. Najčešći je zahtjev da graf aproksimativne funkcije prolazi određenim točkama, odnosno da interpolira funkciju u tim točkama koje nazivamo čvorovi. Kod interpoliranja mogu se postaviti i zahtjevi da se originalna i približna funkcija osim u funkcijskim vrijednostima u čvorovima podudaraju i u nekim vrijednostima derivacija. Drugi čest zahtjev prilikom aproksimiranja je da odstupanje aproksimacijske funkcije od originalne u nekom smislu bude minimalno, to jest da je pogreška aproksimacije čim manja. [6]

Prema tome, imamo dva kriterija određivanja parametara aproksimacije: interpolacija i minimiziranje pogreške. Interpolaciju smo objasnili u uvodu, a kasnije ćemo ju i pobliže obraditi pa ćemo ovdje objasniti samo drugi kriterij aproksimacije.

2.3.1. Minimizacija pogreške

Aproksimativna funkcija φ bira se tako da se minimizira neka odabrana norma funkcije pogreške

$$e(x) = f(x) - \varphi(x),$$

u odabranom vektorskom prostoru funkcija $\mathcal{F}$ definiranih na nekoj domeni X. Takve aproksimacije zovu se najbolje aproksimacije po normi, a dijele se na kontinuirane i diskretne, ovisno o tome minimizira li se norma pogreške e, odnosno $\| e \|$, na kontinuiranom ili diskretnom skupu podataka X. [7]

Kao norme pogreške najčešće se koriste ∞-norma i 2-norma. U slučaju ∞-norme, pripadna aproksimacija zove se minimaks aproksimacija, a za 2-normu pripadna aproksimacija zove se srednjekvadratna i metoda za njezino nalaženje zove se metoda najmanjih kvadrata. U oba slučaja, aproksimacijska funkcija φ, to jest njezini parametri, traže se tako da norma pogreške $\| e \|_2$, odnosno $\| e \|_\infty$ bude minimalna na skupu podataka X. U diskretnom slučaju, skup podataka X definiran je kao $X = \{x_0, \dots, x_n\}$, a u kontinuiranom slučaju definiran je kao $X = [a, b]$. Minimaks aproksimacija poželjniji je tip aproksimacije, ali je tu vrstu i teže izračunati, npr. ako dobijemo problem minimizacije nederivabilne funkcije. [7]

Na kraju možemo napisati koje sve matematičke probleme u teoriji aproksimacije trebamo riješiti: [6]

- Egzistenciju i jedinstvenost rješenja problema aproksimacije,
- Analizu kvalitete dobivene aproksimacije – ponašanje funkcije greške e, kao i sama vrijednost funkcije,
- Konstrukciju algoritama za računanje najbolje aproksimacije,
- Dokaz efikasnosti i točnosti algoritma.

3. Taylorov polinom

U ovom ćemo poglavlju objasniti jednu od formula koja ima veliku primjenu kod aproksimacije funkcija. Ta formula poznata je pod nazivom Taylorova formula, odnosno formula Taylorovog polinoma. Pomoću nje možemo odrediti približnu vrijednost neke funkcije u određenoj točki, dobiti približne vrijednosti derivacija i integrala, kao i približna rješenja nelinearnih jednadžbi.

Za funkciju f, koja je derivabilna n puta, možemo definirati polinom n-tog stupnja

$$P_n(x) = a_0 + a_1(x - x_0) + a_2(x - x_0)^2 + \cdots + a_n(x - x_0)^n$$

sa sljedećim svojstvima:

$$P_n(x_0) = f(x_0), \qquad P_n'(x_0) = f'(x_0), \qquad \dots, \qquad P_n^{(n)}(x_0) = f^{(n)}(x_0).$$

Koeficijenti $a_0, a_1, \dots, a_n$ polinoma P_n tada glase

$$a_0 = f(x_0), \qquad a_1 = f'(x_0), \qquad a_2 = \frac{1}{2!}f''(x_0), \dots, a_n = \frac{1}{n!}f^{(n)}(x_0).$$

Polinom P_n s ovako definiranim koeficijentima nazivamo Taylorov polinom n-tog stupnja u okolini točke x_0 i označavamo ga s T_n:

$$T_n(x) = f(x_0) + f'(x_0)(x - x_0) + \frac{1}{2!}f''(x_0)(x - x_0)^2 + \cdots + \frac{1}{n!}f^{(n)}(x_0)(x - x_0)^n$$

$$= \sum_{k=0}^{n} \frac{f^{(k)}(x_0)}{k!}(x - x_0)^k.$$

Naravno, Taylorov polinom stupnja n možemo izračunati samo ako funkcija f u točki x_0 posjeduje sve derivacije do n-te. [8]

3.1. Greška Taylorovog polinoma

Sa $R_n(x)$ označavamo razliku između točne vrijednosti funkcije $f(x)$ i njezine približne vrijednosti $T_n(x)$, to jest grešku aproksimacije Taylorovim polinomom

$$R_n(x) = f(x) - T_n(x).$$

Grešku $R_n(x)$ možemo izračunati i kao

$$R_n(x) = \frac{f^{(n+1)}(c)}{(n+1)!}(x - x_0)^{n+1},$$

za neki c koji se nalazi između točaka x_0 i x. Obično se povećanjem broja članova, to jest povećanjem stupnja Taylorovog polinoma dobije točnija aproksimacija funkcije f u okolini točke x_0. [8]

Primjer: Greška aproksimacije $f(x) = e^x$ oko nule Taylorovim polinomom

$$T_3(x) = 1 + x + \frac{x^2}{2} + \frac{x^3}{3}$$

iznosi

$$R_3(x) = e^x - 1 - x - \frac{x^2}{2} - \frac{x^3}{3}.$$

Ta greška za primjerice $x = 0.1$ iznosi $R_3(0.1) = 4.2514 \cdot 10^{-6}$. [9]

Zadatak: Rastavite polinom $x^3 - 4x^2 + 10x - 14$ po potencijama od $(x-1)$.

Rješenje:

$$f(x) = x^3 - 4x^2 + 10x - 14$$

$$f(1) = -7$$

$$f'(x) = 3x^2 - 8x + 10$$

$$f'(1) = 5$$

$$f''(x) = 6x - 8$$

$$f''(1) = -2$$

$$f'''(x) = 6$$

$$f'''(1) = 6$$

$$T(x) = \sum_{k=0}^{3} \frac{f^{(k)}(x_0)}{k!}(x - x_0)^k$$

$$\begin{aligned} T(x) &= -7 + 5(x-1) - \frac{2}{2!}(x-1)^2 + \frac{6}{3!}(x-1)^3 \\ &= -7 + 5(x-1) - (x-1)^2 + (x-1)^3 \end{aligned}$$

Zadatak: Odredimo Taylorove polinome u okolini točke $x_0 = 0$ reda $n = 1, 2, 3, 4, 5$ za funkciju $f(x) = \sin x$. [8]

Najprije moramo izračunati derivacije funkcije f u točki $x_0 = 0$.

$$f(x) = \sin x, \qquad f(0) = 0$$

$$f'(x) = \cos x, \qquad f'(0) = 1$$

$$f''(x) = -\sin x, \qquad f''(0) = 0$$

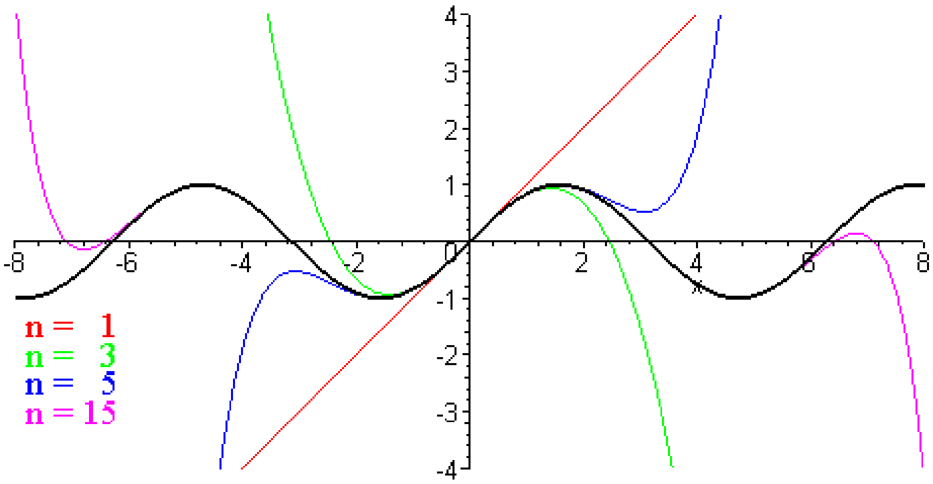
$$f'''(x) = -\cos x, \qquad f'''(0) = -1$$

$$f^{(4)}(x) = \sin x\,, \qquad f^{(4)}(0) = 0$$
$$f^{(5)}(x) = \cos x\,, \qquad f^{(5)}(0) = 1$$

Sada imamo:

$$T_1(x) = T_2(x) = x, \qquad T_3(x) = T_4(x) = x - \frac{1}{3!}x^3, \qquad T_5(x) = x - \frac{1}{3!}x^3 + \frac{1}{5!}x^5.$$

Sljedeća slika prikazuje funkciju $f(x) = \sin x$ oko nule i Taylorove aproksimacije te funkcije polinomom razvijenog do sljedećih stupnjeva: $1, 3, 5$ i 15. Crna boja označava funkciju $\sin x$. Možemo primijetiti da za bolje aproksimacije, a ujedno i manje greške, moramo razviti funkciju do viših stupnjeva i tako se sve više približavamo traženoj funkciji.



Slika 2: Funkcija sin x i Taylorovi polinomi
Izvor: [*https://sh.wikipedia.org/wiki/Tejlorov_polinom#/media/File:Sintay.svg*]

4. Interpolacija polinomima

U uvodu smo opisali što je to interpolacija, a ovdje ćemo, uz pomoć raznih metoda i primjera, pobliže objasniti što je to točno problem interpolacije polinomima. Za određivanje polinoma stupnja n trebamo $n+1$ nezavisan uvjet pa se problem interpolacije definira na sljedeći način:

Neka su zadane točke (x_i, y_i), $i = 0, \dots., n$. Potrebno je pronaći polinom $p(x) = p_n(x)$ n-tog stupnja uz uvjet da je

$$p_n(x_i) = f(x_i), \qquad i = 0, \dots., n.$$

Ako vrijedi $x_i \neq x_j$, za $i \neq j$, onda interpolacijski polinom postoji i on je jedinstven. Međutim, kako u prostoru polinoma postoje različite baze, tako postoje i različiti prikazi interpolacijskog polinoma. Mi razlikujemo tri prikaza: [10]

i) Prikaz u standardnoj bazi
ii) Lagrangeov oblik interpolacijskog polinoma
iii) Newtonov oblik interpolacijskog polinoma

4.1. Prikaz polinoma u standardnoj bazi

Kod tog prikaza interpolacijskog polinoma baza je $\{1, x, x^2, \dots, x^n\}$, a polinom se dobije tako da se riješi sustav [8]

$$p(x_i) = y_i, \qquad i = 0, \dots, n.$$

Zadatak: Pronađite interpolacijski polinom stupnja 2 koji prolazi točkama $(-1,3), (1,5), (2,0)$.
Rješenje:

Znamo da jednadžba polinoma 2. stupnja glasi

$$p(x) = ax^2 + bx + c.$$

Da pronađemo koeficijente a, b, c trebamo riješiti sustav jednadžbi

$$\begin{cases} p(-1) \ = a - b + c = 3 \\ p(1) \quad\ = a + b + c = 5 \\ p(2) = 4a + 2b + c = 0 \end{cases}$$

$$\begin{bmatrix} 1 & -1 & 1 & \vdots & 3 \\ 1 & 1 & 1 & \vdots & 5 \\ 4 & 2 & 1 & \vdots & 0 \end{bmatrix}.$$

Kada prvi redak matrice pomnožimo s -1 i pribrojimo drugom retku, a ujedno prvi redak pomnožimo s -4 i pribrojimo trećem retku dobivamo sljedeći rezultat:

$$\begin{bmatrix} 1 & -1 & 1 & \vdots & 3 \\ 0 & 2 & 0 & \vdots & 2 \\ 0 & 6 & -3 & \vdots & -12 \end{bmatrix}.$$

Vidimo da se iz drugog retka dobije da je $b = 1$. Zatim imamo $-3c = -6b - 12 \Longrightarrow c = 6$ i $a = b - c + 3 \Longrightarrow a = -2$.

Rješavanje ovog sustava bio je relativno složen posao, a da je stupanj polinoma bio veći, on bi bio i još složeniji. Zato se traže druga rješenja za određivanje polinoma.

4.2. Egzistencija i jedinstvenost interpolacijskog polinoma

Ovdje ćemo dokazati teorem koji vrijedi kod interpolacije polinomima. Njegov dokaz koristi činjenicu da postoji jedinstveno rješenje za linearni sustav s regularnom matricom. U teoremu se koristi oznaka $\mathbb{N}_0$ koja označava skup cijelih nenegativnih brojeva. $[11]$

Teorem. *Neka je* $n \in \mathbb{N}_0$. *Za zadane točke* (x_k, f_k) *gdje je* $f_k := f(x_k)$ *za* $k = 0, \dots, n$ *i* $x_i \neq x_j$ *za* $i \neq j$, *postoji jedinstveni (interpolacijski) polinom stupnja najviše* n

$$\varphi(x) \coloneqq p_n(x) = a_0 + a_1 x + \cdots + a_n x^n$$

za kojeg vrijedi

$$p_n(x_k) = f_k, \qquad k = 0, \dots, n.$$

Dokaz:

Neka je $p_n = a_0 + a_1 x + \cdots + a_n x^n$ polinom stupnja najviše n. Uvjete interpolacije zapišemo u obliku

$$\begin{gathered} p_n(x_0) = a_0 + a_1 x_0 + \cdots + a_n x_0^n = f_0 \\ p_n(x_1) = a_0 + a_1 x_1 + \cdots + a_n x_1^n = f_1 \\ \vdots \\ p_n(x_n) = a_0 + a_1 x_n + \cdots + a_n x_n^n = f_n. \end{gathered}$$

Sada je potrebno provjeriti ima li taj sustav od $(n+1)$-e linearne jednadžbe s $(n+1)$-om nepoznanicom $a_0, \dots, a_n$ jedinstveno rješenje. Za to je dovoljno provjeriti je li matrica tog sustava regularna. Provjera se vrši računanjem vrijednosti determinante te matrice, a ta determinanta poznata je pod nazivom Vandermondeova determinanta

$$D_n = \begin{vmatrix} 1 & x_0 & x_0^2 & \cdots & x_0^n \\ 1 & x_1 & x_1^2 & \cdots & x_1^n \\ \vdots & \vdots & \vdots & \cdots & \vdots \\ 1 & x_{n-1} & x_{n-1}^2 & \cdots & x_{n-1}^n \\ 1 & x_n & x_n^2 & \cdots & x_n^n \end{vmatrix}.$$

Definirajmo determinantu koja je slična D_n. Ona umjesto potencija od x_n u posljednjem retku ima potencije od x:

$$V_n(x) = \begin{vmatrix} 1 & x_0 & x_0^2 & \cdots & x_0^n \\ 1 & x_1 & x_1^2 & \cdots & x_1^n \\ \vdots & \vdots & \vdots & \cdots & \vdots \\ 1 & x_{n-1} & x_{n-1}^2 & \cdots & x_{n-1}^n \\ 1 & x & x^2 & \cdots & x^n \end{vmatrix}.$$

Možemo vidjeti da je $D_n = V_n(x_n)$. Ako promatramo $V_n(x)$ kao funkciju od x, razvojem po posljednjem retku uočavamo da je $V_n(x)$ polinom stupnja najviše n u varijabli x, a koeficijent tog polinoma uz x^n je determinanta D_{n-1}. Ako se u determinantu $V_n(x)$ redom uvrsti $x_0, \dots, x_{n-1}$, ta determinanta imat će dva jednaka retka pa će biti

$$V_n(x_0) = V_n(x_1) = \cdots = V_n(x_{n-1}) = 0,$$

odnosno točke $x_0, \dots, x_{n-1}$ su nultočke polinoma $V_n(x)$ n-tog stupnja. Da bi se polinom n-tog stupnja mogao točno odrediti, ako su poznate njegove nultočke, potrebno je još samo znati njegov vodeći koeficijent, a to smo ovdje i pokazali. Vodeći koeficijent je D_{n-1}. Pošto znamo vodeći koeficijent D_{n-1} i sve nultočke $x_0, \dots, x_{n-1}$ $V_n(x)$ možemo zapisati kao produkt

$$V_n(x) = D_{n-1}(x - x_0)(x - x_1) \cdots (x - x_{n-1}).$$

Ako uvrstimo $x = x_n$, dobivamo rekurzivnu formulu za D_n:

$$D_n = D_{n-1}(x_n - x_0)(x_n - x_1) \cdots (x_n - x_{n-1}).$$

Ako znamo da je $D_0 = 1$, što je trivijalno (gledamo lijevi gornji kut determinante), slijedi da je

$$D_n = \prod_{0 \le j < i \le n} (x_i - x_j).$$

S obzirom da je $x_i \neq x_j$ za $i \neq j$, onda je i $D_n \neq 0$, a vrijedi i obrat. Matrica linearnog sustava je regularna i zato postoji jedinstveno rješenje $a_0, \dots, a_n$ za koeficijente polinoma p_n, to jest jedinstveni interpolacijski polinom. [11]

Takav oblik interpolacije obično zovemo Lagrangeova interpolacija, jer tražena funkcija interpolira samo funkcijske vrijednosti zadane funkcije. [11]

4.3. Lagrangeov interpolacijski polinom

Neka je zadana funkcija $f : [a,b] \to \mathbb{R}$ i međusobno različite točke $x_0, x_1, x_2, \dots, x_n$ iz intervala $[a,b]$. Želimo aproksimirati funkciju f polinomom $L_n(x)$ koji u izabranim točkama ima iste vrijednosti kao i ta funkcija. Kad je problem postavljen ovako, on ima mnogo rješenja, ali ako je stupanj polinoma upravo n postoji samo jedno rješenje. Takav polinom, sa svojstvom $L_n(x_k) = y_k$, zove se Lagrangeov polinom. [12]

Neka su nam zadane točke $(x_0, y_0), (x_1, y_1), \dots, (x_n, y_n)$. Najprije moramo definirati polinome $\ell_i \in P_n, i = 0, \dots, n$ gdje je P_n prostor polinoma kao [13]

$$\ell_i(x) = \prod_{\substack{j=0 \\ j\neq i}}^{n} \frac{x - x_j}{x_i - x_j} = \frac{\omega_i(x)}{\omega_i(x_i)},$$

pri čemu je $\omega(x)$ definiran sa

$$\omega(x) = \prod_{j=0}^{n} (x - x_j),$$

a $\omega_i(x)$ sa

$$\omega_i(x) = \prod_{\substack{j=0 \\ j\neq i}}^{n} (x - x_j) = \frac{\omega(x)}{x - x_i}.$$

Dakle, imamo

$$\ell_0(x) = \prod_{\substack{j=0 \\ j\neq 0}}^{n} \frac{x - x_j}{x_0 - x_j} = \frac{(x - x_1)(x - x_2) \dots (x - x_n)}{(x_0 - x_1)(x_0 - x_2) \dots (x_0 - x_n)}$$

$$\ell_1(x) = \prod_{\substack{j=0 \\ j\neq 1}}^{n} \frac{x - x_j}{x_1 - x_j} = \frac{(x - x_0)(x - x_2) \dots (x - x_n)}{(x_1 - x_0)(x_1 - x_2) \dots (x_1 - x_n)}$$

$$\vdots$$

$$\ell_n(x) = \prod_{\substack{j=0 \\ j\neq n}}^{n} \frac{x - x_j}{x_n - x_j} = \frac{(x - x_0)(x - x_1) \dots (x - x_{n-1})}{(x_n - x_0)(x_n - x_1) \dots (x_n - x_{n-1})}.$$

Polinome $\ell_i,\ i = 0, \dots, n$ zovemo Lagrangeovim koeficijentima ili funkcijama Lagrangeovih baza. Ti polinomi moraju zadovoljiti uvjete

$$\ell_i(x_j) = \delta_{ij} = \begin{cases} 1\ za\ j = i, \\ 0\ za\ j \neq i. \end{cases}$$

gdje je δ_{ij} Kroneckerov simbol, pa polinom $L_n(x)$ koji rješava problem interpolacije možemo zapisati kao

$$L_n(x) = \sum_{i=0}^{n} y_i \ell_i(x)$$

ili u obliku

$$L_n = \sum_{k=0}^{n} \frac{(x - x_0)(x - x_1) \dots (x - x_{k-1})(x - x_{k+1}) \dots (x - x_n)}{(x_k - x_0)(x_k - x_1) \dots (x_k - x_{k-1})(x_k - x_{k+1}) \dots (x_k - x_n)} y_k$$

pri čemu je $y_k = f(x_k)$. [2]

Ovako zapisan polinom nazivamo Lagrangeov interpolacijski polinom, ali precizniji naziv bio bi Lagrangeov oblik interpolacijskog polinoma pošto je interpolacijski polinom jedinstven.

Primjer: Pronađite Lagrangeov oblik interpolacijskog polinoma drugog stupnja koji prolazi točkama $(-3, -4), (4, 2)$ i $(3, 0)$.

Rješenje:

Najprije se trebaju definirati Lagrangeovi koeficijenti ℓ_i pa tako imamo

$$\ell_0(x) = \frac{(x-4)(x-3)}{(-3-4)(-3-3)} = \frac{x^2 - 7x + 12}{42},$$

$$\ell_1(x) = \frac{(x+3)(x-3)}{(4+3)(4-3)} = \frac{x^2 - 9}{7},$$

$$\ell_2(x) = \frac{(x+3)(x-4)}{(3+3)(3-4)} = -\frac{x^2 - x - 12}{6},$$

pa je traženi interpolacijski polinom

$$L_n(x) = (-4)\left(\frac{x^2 - 7x + 12}{42}\right) + 2\frac{x^2 - 9}{7} + 0\left(-\frac{x^2 - x - 12}{6}\right) = \frac{4}{21}x^2 + \frac{2}{3}x - \frac{26}{7}.$$

Kod rješavanja zadataka s Lagrangeovim interpolacijskim polinomom, koristit ćemo alat MATLAB pomoću kojeg se taj polinom lako odredi i prikaže grafički. MATLAB posjeduje funkciju $polyfit$ čije su ulazne varijable jednoredne matrice x i y koje sadrže sve apscise i ordinate zadanih točaka, kao i prirodni broj k koji označava stupanj traženog polinoma. Izlazna varijabla je matrica L čiji su elementi svi koeficijenti polinoma stupnja najviše k. Kako bismo kao rezultat dobili Lagrangeov interpolacijski polinom, uzet ćemo da je $k = n - 1$. Ovdje ćemo koristiti i funkciju $polyval$ koja računa vrijednost polinoma s koeficijentima zadanima jednorednom matricom p u svim točkama od x. $[14]$

Primjer: Odredimo Lagrangeov interpolacijski polinom čiji graf prolazi točkama $T_1 = (-1, 1), T_2 = (2, -2), T_3 = (-3, -1.5), T_4 = (-4, 2.5)$ i $T_5 = (5, 4)$ i prikažimo ga grafički na segmentu $[-6, 6]$. [14]

Najprije moramo zadati jednoredne matrice x i y. U komandni prozor upišemo:

$x = [-1 \quad 2 \quad -3 \quad -4 \quad 5];$

$y = [1 \quad -2 \quad -1.5 \quad 2.5 \quad 4];$

Sada određujemo matricu L upisivanjem:

$L = polyfit\ (x, y, 4)$

Vrijednost 4 smo unijeli jer za $n = 5$ točaka imamo $k = n - 1 = 4$. Kad pritisnemo $Enter$, MATLAB ispisuje:

$L =$

$Columns\ \ 1\ \ through\ \ 4$

$0.59393518518519 \quad -0.04305555555556 \quad -1.02152777777778$

-0.11898148148148

$Column\ \ 5$

1.80555555555558

Dakle, traženi polinom je oblika

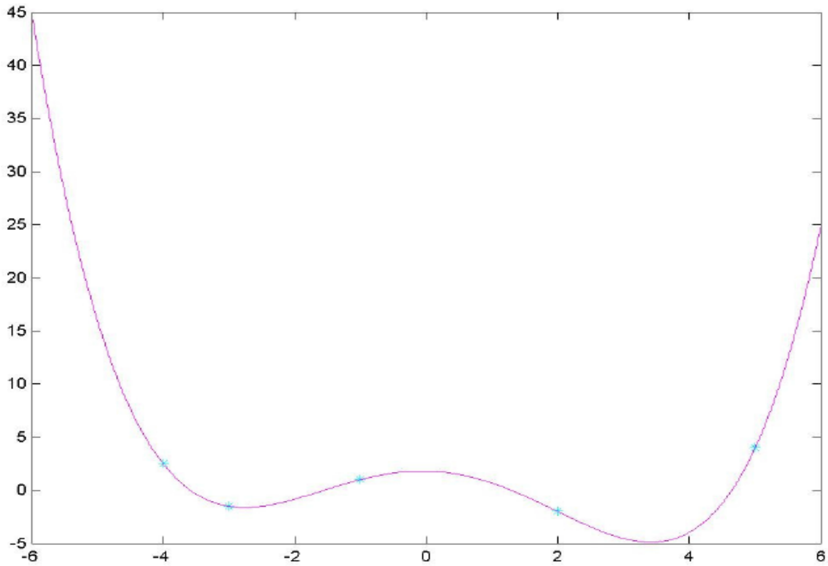
$$L(x) = 0.59393518518519\, x^4 - 0.04305555555556\, x^3 - 1.02152777777778\, x^2 - 0.11898148148148x + 1.80555555555558.$$

Koeficijenti tog polinoma su decimalni brojevi budući da MATLAB ne može računati s razlomcima. Prikažimo sada dobiveni polinom grafički. U komandnom prozoru upišemo:

$x1 = -6 : 0.01\ : 6;$

$plot\ (x, y,\ '*', x1, polyval\ (L, x1))$

U posljednjoj smo naredbi zapisali da na istoj slici želimo ucrtati zadane točke (označili smo ih znakom *), kao i interpolacijski polinom na zadanom segmentu. Dobije se sljedeći graf:



Slika 3: Lagrangeov interpolacijski polinom u MATLAB-u
Izvor: [*14*]

4.3.1. Greška Lagrangeovog interpolacijskog polinoma

Kada se radi aproksimacija funkcije $f(x)$ pomoću Lagrangeovog interpolacijskog polinoma tada se napravi neka greška zato jer umjesto prave vrijednosti funkcije $f(x)$ uzimamo da je $L_n(x) \approx f(x)$. Da bi odredili tu grešku trebamo Rolleov teorem koji kaže: ako je funkcija g klase $C^1(a,b)$ i $g(a) = g(b)$ onda postoji točka $\xi \in (a,b)$ takva da je $g'(\xi) = 0$. Dakle, prva derivacija funkcije g poništava se barem u jednoj točki iz intervala (a,b). [5]

Sada je potrebno definirati funkciju

$$e(t) = f(t) - L_n(t),$$

gdje je $L_n(t)$ Lagrangeov interpolacijski polinom. Pretpostavit ćemo da je $f \in C^{(n+1)}(a,b)$. Zatim se definira

$$g(x) = e(x) - \frac{\omega_n(x)}{\omega_n(t)} e(t),$$

gdje je $\omega_n(x) = (x - x_0)(x - x_1) \dots (x - x_n) = \prod_{k=0}^{n}(x - x_k)$. Kako su funkcije e i ω_n $(n+1)$-puta neprekidno diferencijabilne, tako je i funkcija g. Osim toga je

$$g(x_i) = e(x_i) - \frac{\omega_n(x_i)}{\omega_n(t)} e(t) = 0, \qquad i = 0, 1, 2, \dots, n,$$

jer je $\omega_n(x_i) = 0$ i $e(x_i) = f(x_i) - L_n(x_i) = 0, i = 0, 1, 2, \dots, n$. Isto tako vrijedi i

$$g(t) = e(t) - \frac{\omega_n(t)}{\omega_n(t)} e(t) = 0.$$

Prethodne dvije relacije nam pokazuju da se funkcija g poništava u $(n+1)+1=n+2$ točke $(t, x_0, x_1, \dots, x_n)$. Kad se primijeni Rolleov teorem na tu funkciju vidi se da se prva derivacija funkcije g poništava u barem $(n+1)$-oj točki. Primjenom Rolleovog teorema na prvu derivaciju vidi se da se druga derivacija funkcije g poništava u barem n točaka. Ako nastavimo dalje, vidimo da se $(n+1)$-va derivacija funkcije g poništava u barem jednoj točki unutar intervala (a, b). Tu točku označimo s ξ,

$$g^{(n+1)}(\xi)=0.$$

S druge strane se vidi da je

$$g^{(n+1)}(x)=e^{(n+1)}(x)-\frac{\omega_n^{(n+1)}(x)}{\omega_n(t)}e(t)=f^{(n+1)}(x)-\frac{(n+1)!}{\omega_n(t)}e(t),$$

jer je $L_n^{(n+1)}(x)=0$ i $\omega_n^{(n+1)}(x)=(n+1)!$, pa je

$$g^{(n+1)}(\xi)=f^{(n+1)}(\xi)-\frac{(n+1)!}{\omega_n(t)}e(t)=0.$$

Iz prethodne relacije se dobije

$$e(t)=\frac{\omega_n(t)}{(n+1)!}f^{(n+1)}(\xi), \qquad \xi \in (a, b).$$

Tako je

$$f(x)=L_n(x)+\frac{\omega_n(x)}{(n+1)!}f^{(n+1)}(\xi).$$

Iz prethodne relacije član

$$\frac{\omega_n(x)}{(n+1)!}f^{(n+1)}(\xi)$$

označava grešku interpolacije. Ako je

$$M_{n+1} \coloneqq \max_{x \in [a,b]} \left|f^{(n+1)}(x)\right|,$$

ta greška može se ocijeniti sa

$$|e(x)| \leq \frac{|\omega_n(x)|}{(n+1)!}M_{n+1}.$$

Može se zaključiti kako se povećanjem broja točaka (n) smanjuje pogreška (radi $1/(n+1)!$). Ova ocjena greške vrijedi i za Newtonov oblik interpolacijskog polinoma kojeg ćemo objasniti malo kasnije. [5]

Zadatak: Potrebno je konstruirati interpolacijski polinom $L_2(x)$ za funkciju $f(x)=\sqrt{x}$ ako su zadani čvorovi interpolacije $x_0=100$, $x_1=144$, $x_2=196$. Također, treba izračunati $L_2(150)$ i ocijeniti grešku u točki $x=150$.

Rješenje:

Kad uvrstimo zadane čvorove interpolacije u funkciju $f(x) = \sqrt{x}$ dobijemo sljedeću tablicu:

Tablica 1: Funkcija $f(x) = \sqrt{x}$

x	100	144	196
$f(x)$	10	12	14

Lagrangeov interpolacijski polinom za funkciju $f(x) = \sqrt{x}$ je oblika:

$$L_2(x) = \frac{(x-144)(x-196)}{(100-144)(100-196)} \cdot 10 + \frac{(x-100)(x-196)}{(144-100)(144-196)} \cdot 12$$
$$+ \frac{(x-100)(x-144)}{(196-100)(196-144)} \cdot 14$$
$$= -\frac{2}{27456}x^2 + \frac{1736}{27456}x + \frac{120960}{27456}.$$
$$\Rightarrow L_2(150) = 15.528846.$$

Derivacije funkcije $f(x) = \sqrt{x}$ su:

$$f'(x) = \frac{1}{2\sqrt{x}}, \qquad f''(x) = -\frac{1}{4x^{3/2}}, \qquad f'''(x) = \frac{3}{8x^{5/2}}$$

Maksimum apsolutne vrijednosti treće derivacije funkcije je:

$$M_3 = \max_{[100,196]} |f'''(x)| = \frac{3}{8 \cdot 100^{5/2}} = \frac{3}{8} \cdot 10^{-5},$$

a greška interpolacije u točki 150 je:

$$|R_2(150)| \leq \frac{\frac{3}{8} \cdot 10^{-5}}{3!} |(150-100)(150-144)(150-196)| \leq 8.63 \cdot 10^{-3}$$

Zadatak:

a) Potrebno je pronaći interpolacijski polinom koji funkciju $f(x) = \sin(\pi x)$ interpolira u točkama s x-koordinatama:

$$x_0 = 0, x_1 = \frac{1}{2}, x_2 = 1$$

b) Zatim treba ocijeniti grešku dobivene interpolacije,

c) Izračunati vrijednost dobivenog interpolacijskog polinoma u točki $x = 0.3$,

d) Ocijeniti grešku interpolacije točki $x = 0.3$ i

e) Naći pravu grešku

Rješenje:

a) Najprije moramo uvrstiti $x_0 = 0, x_1 = \frac{1}{2}$ i $x_2 = 1$ u funkciju $f(x) = \sin(\pi x)$. Dobivamo rezultate $y_0 = 0, y_1 = 1$ i $y_2 = 0$. Pošto imamo tri podatka, potrebno je naći interpolacijski polinom $L_2(x)$ drugog stupnja. Računajući dobivamo Lagrangeov interpolacijski polinom koji glasi:

$$L_2(x) = -\frac{x^2}{4} + \frac{x}{4}$$

b) Za ocjenu greške treba nam $\omega(x)$

$$\omega(x) = (x - x_0)(x - x_1)(x - x_2) = \cdots = x^3 - \frac{3}{2}x^2 + \frac{1}{2}x,$$

kao i treća derivacija funkcije f

$$f(x) = \sin(\pi x)$$
$$f'(x) = \pi \cos(\pi x)$$
$$f''(x) = -\pi^2 \sin(\pi x)$$
$$f^{(3)}(x) = -\pi^3 \cos(\pi x).$$

Sada imamo

$$M_3 = \max_{x \in [0,1]} \left| f^{(3)}(x) \right| = \max_{x \in [0,1]} \left| -\pi^3 \cos(\pi x) \right| = \pi^3 \max_{x \epsilon [0,1]} \left| \cos(\pi x) \right| = \pi^3$$

jer je $0 \in [0,1]$ i $\cos 0 = 1$.

Sukladno tome, ocjena greške za neku točku iz intervala $[0,1]$ glasi

$$|f(x) - L_2(x)| \leq \frac{\pi^3}{3!} \left| x^3 - \frac{3}{2}x^2 + \frac{1}{2}x \right|.$$

c) Vrijednost interpolacijskog polinoma u zadanoj točki iznosi

$$L_2(0.3) = -\frac{(0.3)^2}{4} + \frac{0.3}{4} = 0.0525.$$

d) Ocjena greške (budući da je $0.3 \in [0,1]$) iznosi

$$|f(0.3) - L_2(0.3)| \leq \frac{\pi^3}{3!} \left| (0.3)^3 - \frac{3}{2}(0.3)^2 + \frac{1}{2} \cdot 0.3 \right| = 0.217043936.$$

e) Prava greška iznosi

$$f(0.3) - L_2(0.3) = \sin(\pi \cdot 0.3) - 0.0525 = 0{,}756516994$$

Lagrangeov interpolacijski polinom ima i svoje nedostatke. Najveći od njih je taj što postoji velik broj aritmetičkih operacija kod određivanja interpolacijskog polinoma i ako se žele dodati ili izbaciti neke točke nakon što smo već odredili interpolacijski polinom, potrebno je cijeli izračun raditi ispočetka, bez mogućnosti da se koriste prethodno izračunati rezultati. Zato se Lagrangeov oblik interpolacijskog polinoma više koristi u teoriji (za dokaze), a u praksi se koristi Newtonov oblik interpolacijskog polinoma. Isto tako, ako se odabere polinom preniskog stupnja, on može slabo aproksimirati neko rješenje, to jest originalnu funkciju, a ako se odabere polinom previsokog stupnja, onda se mogu pojaviti velike oscilacije, to jest greške na rubovima intervala aproksimacije.

Lagrangeov interpolacijski polinom može se koristiti i u druge svrhe. Na primjer, pomoću njega se može odrediti približna vrijednost derivacije funkcije u nekoj točki iz intervala $[a, b]$. U praksi se često želi pronaći određena vrijednost funkcije pomoću linearne interpolacije. Za to nam služi Lagrangeov interpolacijski polinom $L_1(x)$. [5]

4.3.2. Lagrangeov interpolacijski polinom za ekvidistantne čvorove

Ekvidistantna mreža predstavlja raspored čvorova kod koje su uzastopni čvorovi jednako udaljeni, to jest razmak između čvorova je konstantan. Taj razmak zove se korak interpolacije i označava se s $h = x_{i+1} - x_i,\ i = 0, 1, 2, \ldots, n-1$. Čvorove x_i u ekvidistantnoj mreži možemo zapisati i kao $x_i = x_0 + ih$, a korak interpolacije može se još izračunati kao $h = \frac{x_n - x_0}{n}$. Uvedemo varijablu [17]

$$t = \frac{x - x_0}{h}$$

pa tako imamo:

$$x - x_0 = th, \qquad x = x_0 + th,$$
$$x - x_1 = (x - x_0) - (x_1 - x_0) = th - h$$
$$x_i - x_0 = ih, \qquad x_i - x_1 = (i-1)h$$

itd., pa možemo pisati

$$\frac{(x - x_0)(x - x_1)\cdots(x - x_{i-1})(x - x_{i+1})\cdots(x - x_n)}{(x_i - x_0)(x_i - x_1)\cdots(x_i - x_{i-1})(x_i - x_{i+1})\cdots(x_i - x_n)} =$$
$$= \frac{th(th - h)\cdots[th - (i-1)h][th - (i+1)h]\cdots(th - nh)}{ih(i-1)h\cdots h(-h)\cdots[-(n-i)h]}$$
$$= \frac{t(t-1)\cdots(t-n)}{t-i}\cdot\frac{(-1)^{n-i}}{i!\,(n-i)!}$$
$$= \frac{t(t-1)\cdots(t-n)}{t-i}\cdot\frac{(-1)^{n-i}}{n!}\cdot\frac{n!}{i!\,(n-i)!}$$
$$= (-1)^{n-1}\binom{n}{i}\cdot\frac{1}{t-i}\cdot\frac{t(t-1)\cdots(t-n)}{n!}$$

Time dobivamo Lagrangeov oblik interpolacijskog polinoma za ekvidistantne čvorove

$$L_n(x) = L_n(x_0 + th) = (-1)^n \frac{t(t-1)\cdots(t-n)}{n!} \sum_{i=0}^{n} (-1)^i \binom{n}{i} \frac{f(x_i)}{t-i} .$$

Ocjena pogreške interpolacijskog polinoma stupnja n na ekvidistantnoj mreži s korakom h glasi

$$|f(x) - p(x)| \leq \frac{h^{n+1}}{n+1} M_{n+1},$$

gdje je, kao što već otprije znamo,

$$M_{n+1} \coloneqq \max_{x \in [a,b]} \left| f^{(n+1)}(x) \right|.$$

4.4. Newtonov oblik interpolacijskog polinoma

Cilj u ovom poglavlju je odrediti Newtonov oblik interpolacijskog polinoma za neekvidistantne, kao i za ekvidistantne čvorove. Najprije ćemo se baviti Newtonovim oblikom polinoma za neekvidistantne čvorove, a u tu svrhu uvodimo pojam podijeljenih razlika, koje se definiraju rekurzivno na sljedeći način.

Podijeljena razlika nultog reda $f[x_i]$ je po definiciji vrijednost funkcije f u točki $x_i \in \{x_0, x_1, \dots, x_n\}$, odnosno $f[x_i] = f(x_i)$. Podijeljene razlike prvog reda definiramo sa

$$f[x_i, x_j] = \frac{f(x_j) - f(x_i)}{x_j - x_i}.$$

Podijeljene razlike drugog reda definiramo sa

$$f[x_i, x_j, x_k] = \frac{f[x_j, x_k] - f[x_i, x_j]}{x_k - x_i}.$$

Na kraju, podijeljenu razliku n- tog reda definiramo sa

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

Sada ćemo formirati tablicu kako bi vidjeli kako se podijeljene razlike mogu praktično računati.[5]

Tablica 2: Tablica podijeljenih razlika

x_k	$f(x_k) = f[x_k]$	$f[x_k, x_{k+1}]$	$f[x_k, x_{k+1}, x_{k+2}]$	$\dots$	$f[x_0, \dots, x_n]$
x_0	$\boldsymbol{f[x_0]}$				
		$\boldsymbol{f[x_0, x_1]}$			
x_1	$f[x_1]$		$\boldsymbol{f[x_0, x_1, x_2]}$		
				$\ddots$	
$\vdots$	$\vdots$	$\vdots$	$\vdots$		$\boldsymbol{f[x_0, \dots, x_n]}$
				$⋰$	
x_{n-1}	$f[x_{n-1}]$		$f[x_{n-2}, x_{n-1}, x_n]$		
		$f[x_{n-1}, x_n]$			
x_n	$f[x_n]$				

Newtonov interpolacijski polinom stupnja n glasi

$$p_n(x) = a_0 + a_1(x - x_0) + \cdots + a_n \prod_{k=0}^{n-1}(x - x_k),$$

a koeficijenti $a_k,\ k = 0, 1, \dots, n$ mogu se definirati kao podijeljene razlike $a_k = f[x_0, \dots, x_k]$. Ako koeficijente u formuli za Newtonov interpolacijski polinom zamijenimo podijeljenim razlikama prema tablici koju smo prethodno formirali, Newtonov oblik interpolacijskog polinoma izgleda ovako:

$$p_n(x) = f[x_0] + f[x_0, x_1](x - x_0) + f[x_0, x_1, x_2](x - x_0)(x - x_1) + \cdots +$$
$$+ f[x_0, \dots, x_n](x - x_0)(x - x_1) \cdots (x - x_{n-1}).$$

U tablici podijeljenih razlika podebljali smo neke članove. Ti članovi označavaju „gornji rub“, i oni su nam najbitniji kod izračuna Newtonovog interpolacijskog polinoma. On se može izračunati u jednom jednodimenzionalnom polju. [18]

4.4.1. Algoritam za izračun podijeljenih razlika

Srž algoritma za izračun podijeljenih razlika, to jest „gornjeg ruba“ tablice podijeljenih razlika glasi:

$za\ i = 1\ do\ n\ radi\ \{$

$\quad za\ j = n\ do\ i\ radi\ \{$

$\qquad f[j] = (f([j] - f[j-1])/(x[j] - x[j-i]);$

$\quad \};$

$\};$

Kad algoritam završi, „gornji rub“ $f[x_0, \dots, x_i]$ se nalazi, redom, u polju f. [18]

4.4.2. Greška Newtonovog interpolacijskog polinoma

Iako se greška koju smo pokazali za Lagrangeov oblik interpolacijskog polinoma može koristiti i za Newtonov oblik, grešku za Newtonov oblik interpolacijskog polinoma možemo pokazati i pomoću podijeljenih razlika.

Neka je x_{n+1} realan broj koji ne pripada skupu čvorova $\{x_0, x_1, \dots, x_n\}$. Konstruiramo novi interpolacijski polinom p_{n+1} koji prolazi čvorovima $x_0, \dots, x_{n+1}$:

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

Budući da je

$$p_{n+1}(x_{n+1}) = f(x_{n+1})$$

imamo

$$f(x_{n+1}) = p_n(x_{n+1}) + (x_{n+1} - x_0)\cdots(x_{n+1} - x_n)f[x_0, x_1, \dots, x_{n+1}].$$

Ako se ovaj izraz usporedi s ocjenom greške općeg interpolacijskog polinoma

$$f(x_{n+1}) - p_n(x_{n+1}) = \frac{\omega_n(x_{n+1})}{(n+1)!} f^{(n+1)}(\xi),$$

gdje je ξ neki proizvoljni broj između x_0 i x_{n+1}, a $\omega_n = (x - x_0)(x - x_1)\cdots(x - x_n)$, slijedi da je

$$f[x_0, x_1, \dots, x_n, x_{n+1}] = \frac{f^{(n+1)}(\xi)}{(n+1)!},$$

odnosno (jer je x_{n+1} proizvoljan)

$$f[x_0, x_1, \dots, x_n, x] = \frac{f^{(n+1)}(\xi)}{(n+1)!}.$$

Na kraju se dobije

$$f(x) - p_n(x) = \omega_n(x) f[x_0, x_1, \dots, x_n, x].$$

Ako pretpostavimo da se u polju f na mjestu i nalazi $f[x_0, x_1, \dots, x_i]$, dobivamo algoritam za izvrednjavanje polinoma $p_n(x)$. [19]

4.4.3. Algoritam za izvrednjavanje $p_n(x)$

Srž algoritma za izvrednjavanje polinoma $p_n(x)$ u točki x ima oblik Hornerove sheme i glasi:

$sum = f[n];$

$za\ i = n - 1\ do\ 0\ radi\ \{$

$$sum = sum \cdot (x - x[i]) + f[i];$$

$\};$

Na kraju se dobije da je $p_n(x) = sum$. [18]

Zadatak: Pronađite interpolacijski polinom drugog stupnja od funkcije $f(x) = \ln(1 + x)$ sa čvorovima interpolacije $0, 1, 3$.

a) Nađite $p_n(2)$, pravu grešku i ocjenu prave greške u točki 2,

b) Napravite ocjenu pogreške interpolacijskog polinoma

Rješenje:

Najprije napravimo tablicu podijeljenih razlika

Tablica 3: Tablica podijeljenih razlika za $f(x) = ln(1 + x)$

x_k/k	0	1	2
0	$\mathbf{0}$		
		$\mathbf{\ln 2}$	
1	$\ln 2$		$\dfrac{\frac{\ln 2}{2} - \ln 2}{3 - 0} = -\mathbf{\dfrac{\ln 2}{6}}$
		$\dfrac{\ln 4 - \ln 2}{3 - 1} = \dfrac{\ln 2}{2}$	
3	$\ln 4$		

Dakle, imamo

$$p_n(x) = 0 \cdot 1 + \ln 2\,(x - 0) + \left(-\frac{\ln 2}{6}\right)(x - 0)(x - 1)$$

$$= x \ln 2 - \frac{\ln 2}{6} x(x - 1).$$

a)

Polinom $p_n(2)$ iznosi

$$p_n(2) = \frac{5}{3} \ln 2.$$

Prava pogreška iznosi

$$|f(2) - p_n(2)| = \left|\ln 2 - \frac{5}{3} \ln 2\right| = 0.056633.$$

b) Da bi našli ocjenu pogreške, trebamo treću derivaciju funkcije f i njen maksimum na intervalu $[0, 3]$.

$$f'(x) = \frac{1}{x+1}, \qquad f''(x) = -\frac{1}{(x+1)^2}, \qquad f'''(x) = \frac{2}{(x+1)^3},$$

Kako je $f'''(x)$ padajuća i strogo pozitivna funkcija na intervalu $[0, 3]$, njen maksimum se postiže u točki 0 pa je

$$M_{n+1} = \max_{x \in [0,3]} |f'''(x)| = |f'''(0)| = 2,$$

$$\Rightarrow |f(2) - p_n(2)| \leq \frac{|(2-0)(2-1)(2-3)|}{(2+1)!} \cdot 2 = \frac{2}{3} = 0.667.$$

Vidimo da je ova ocjena greške puno gora od prave pogreške.

Zadatak: Potrebno je aproksimirati funkciju $f(x) = e^x$ na segmentu $[0, 0.5]$ interpolacijskim polinomom. U zadatku ćemo koristiti Lagrangeov i Newtonov oblik interpolacijskog polinoma te ćemo vidjeti prednosti, to jest nedostatke pojedinog oblika. Pri tome ćemo sve rezultate zaokružiti na 6 decimala. Funkciju $f(x) = e^x$ aproksimiramo na osnovu sljedećih podataka: [20]

Tablica 4: Funkcija $f(x) = e^x$

k	0	1	2
x_k	0.0	0.2	0.5
$f(x_k)$	1.000000	1.221403	1.648721

Lagrangeov interpolacijski polinom drugog stupnja (pošto imamo tri podatka) glasi

$$L_2(x) = 1\frac{(x-0.2)(x-0.5)}{(0-0.2)(0-0.5)} + 1.221403\frac{(x-0)(x-0.5)}{(0.2-0)(0.2-0.5)} +$$

$$+1.648721\frac{(x-0)(x-0.2)}{(0.5-0)(0.5-0.2)} = 0.634756x^2 + 0.980064x + 1$$

Da bi zapisali Newtonov oblik interpolacijskog polinoma moramo formirati tablicu podijeljenih razlika:

Tablica 5: Tablica podijeljenih razlika za funkciju $f(x) = e^x$

k	$f[x_k]$	$f[x_k, x_{k+1}]$	$f[x_k, x_{k+1}, x_{k+2}]$
0	1.000000		
		1.107015	
1	1.221403		0.634756
		1.424393	
2	1.648721		

pa Newtonov oblik interpolacijskog polinoma glasi

$$P_2(x) = 1 + 1.107015(x - 0) + 0.634756(x - 0)(x - 0.2)$$
$$= 0.634756x^2 + 0.980064x + 1$$

Pokazali smo da je interpolacijski polinom jedinstven. Prema tome bi Lagrangeov i Newtonov oblik interpolacijskog polinoma trebao biti jednak. Međutim, njihovim uspoređivanjem vidimo da se koeficijenti uz x^2 razlikuju za 10^{-6}. To je zbog grešaka koje nastaju prilikom zaokruživanja u izračunu.

S obzirom da je $f^{(k)}(x) = (e^x)^{(k)} = e^x, k = 1, 2, ...$, greška je

$$|f(x) - P_2(x)| \leq \frac{M_{n+1}}{3!}|x(x - 0.2)(x - 0.5)| \qquad (0 \leq x \leq 0.5)$$

gdje je

$$M_{n+1} = \max_{x \in [0,0.5]} |e^x| = e^{0.5} \cong 1.648721$$

Ako uvedemo novi čvor možemo smanjiti grešku. Stoga uvodimo $x_3 = 0.4$ pa je $f(x_3) = 1.491825$. Da koristimo Lagrangeov oblik interpolacijskog polinoma, morali bi cijeli izračun početi ispočetka, pa ćemo ovdje koristiti Newtonov oblik interpolacijskog polinoma. Korištenjem Newtonove interpolacije imamo

$$P_3(x) = P_2(x) + f[x_0, x_1, x_2, x_3](x - x_0)(x - x_1)(x - x_2).$$

Sada moramo dopuniti tablicu podijeljenih razlika s novim čvorom x_3:

Tablica 6: Tablica podijeljenih razlika za funkciju $f(x) = e^x$ s novim čvorom

k	$f[x_k]$	$f[x_k, x_{k+1}]$	$f[x_k, x_{k+1}, x_{k+2}]$	$f[x_k, x_{k+1}, x_{k+2}, x_{k+3}]$
0	1.000000			
		1.107015		
1	1.221403		0.634756	
		1.424393		0.220198
2	1.648721		0.722835	
		1.568960		
3	1.491825			

Temeljem tablice imamo

$$P_3(x) = P_2(x) + 0.220198x(x - 0.2)(x - 0.5)$$
$$= 0.220198x^3 + 0.480618x^2 + 1.002084x + 1.$$

Greška je

$$|f(x) - P_3(x)| \leq \frac{M_{n+1}}{4!}|x(x - 0.2)(x - 0.5)(x - 0.4)| \qquad (0 \leq x \leq 0.5)$$

Recimo, za $x = 0.3$ imamo

$$|f(0.3) - P_2(0.3)| = 0.001288$$
$$|f(0.3) - P_3(0.3)| = 0.000033$$

4.4.4. Newtonov interpolacijski polinom za ekvidistantne čvorove

Da bi zapisali Newtonov oblik interpolacijskog polinoma za ekvidistantne čvorove moramo najprije objasniti pojam konačnih razlika. Kad imamo ekvidistantne čvorove koji su ravnomjerno raspoređeni s korakom h, $x_i = x_0 + ih$, $i = 0,1, \dots, n$, umjesto podijeljenih razlika koristimo konačne razlike. Postoje konačne razlike unaprijed i konačne razlike unazad.

Konačnu razliku prvog reda unaprijed Δy_i definiramo kao

$$\Delta y_i = y_{i+1} - y_i = \Delta^0 y_{i+1} - \Delta^0 y_i, \qquad i = 0, 1, \dots, n - 1$$

gdje je $y_i = f(x_i)$, $y_{i+1} = f(x_i + h)$. Konačne razlike drugog reda $\Delta^2 y_i$ mogu se dobiti pomoću konačnih razlika prvog reda pa tako imamo

$$\Delta^2 y_i = \Delta(\Delta y_i) = \Delta(y_{i+1} - y_i) = \Delta y_{i+1} - \Delta y_i = (y_{i+2} - y_{i+1}) - (y_{i+1} - y_i)$$
$$= y_{i+2} - 2y_{i+1} + y_i.$$

Ako nastavimo s istim postupkom dobili bi konačne razlike trećeg reda, itd. Vidimo da se svaka konačna razlika može dobiti pomoću konačne razlike reda ispred nje pa tako imamo razliku k-tog reda:

$$\Delta^k y_i = \Delta^{k-1} y_{i+1} - \Delta^{k-1} y_i, \qquad k = 0, 1, \dots, n, \qquad i = 0, 1, \dots, n-k.$$

Za izračun konačnih razlika koristimo dvije tablice: horizontalnu i dijagonalnu. I jedna i druga tablica sadrže iste podatke, ali je njihov način prikazivanja tih podataka različit. Ovdje ćemo prikazati obje tablice. Za danu tablicu s $(n+1)$ članova, možemo izračunati $(n-1)$ konačnu razliku 1. reda, $(n-2)$ konačne razlike 2. reda,.., 2 konačne razlike $(n-1)$-og reda i jednu konačnu razliku n-tog reda. [6]

Tablica 7: Horizontalna tablica konačnih razlika

x_i	y_i	Δy_i	$\Delta^2 y_i$	$\Delta^3 y_i$	$\cdots$	$\Delta^k y_i$
x_0	y_0	Δy_0	$\Delta^2 y_0$	$\Delta^2 y_0$	$\cdots$	$\Delta^k y_0$
x_1	y_1	Δy_1	$\Delta^2 y_1$	$\vdots$	$\vdots$	
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\Delta^2 y_{n-3}$		
			$\Delta^2 y_{n-2}$			
$\vdots$	$\vdots$	Δy_{n-1}				
x_n	y_n					

Tablica 8: Dijagonalna tablica konačnih razlika

x_i	y_i	Δy_i	$\Delta^2 y_i$	$\cdots$	$\Delta^k y_i$
x_0	y_0				
		Δy_0			
x_1	y_1		$\Delta^2 y_0$		
		Δy_1		$\ddots$	
$\vdots$	$\vdots$	$\vdots$	$\vdots$		$\Delta^k y_0$
		Δy_{n-2}		$\iddots$	
x_{n-1}	y_{n-1}		$\Delta^2 y_{n-2}$		
		Δy_{n-1}			
x_n	y_n				

Za iste tablice možemo definirati konačne razlike unazad. To radimo na sljedeći način.

$$\nabla^0 y_i = y_i, \qquad i = 0, 1, \dots, n$$

$$\nabla^1 y_1 = y_i - y_{i-1} = \nabla^0 y_i - \nabla^0 y_{i-1}, \qquad i = 1, 2, \dots, n$$

$$\vdots$$

$$\nabla^k y_i = \nabla^{k-1} y_i - \nabla^{k-1} y_{i-1}, \qquad i = k, \dots, n \ \ (k = 2, 3, \dots, n).$$

Vidimo da je

$$\Delta y_i = y_{i+1} - y_i = \nabla y_{i+1}.$$

Veza između konačnih razlika unaprijed i konačnih razlika unazad može se definirati kao [21]

$$\Delta^k y_i = \nabla^k y_{i+k}, \qquad k = 0, 1, \dots, n, \qquad i = 0, 1, \dots, n-k.$$

4.4.4.1. Prvi Newtonov interpolacijski polinom

Uz pomoć tablice konačnih razlika vidimo da su koeficijenti kod Newtonovog oblika interpolacijskog polinoma jednaki

$$a_0 = y_0, \qquad a_1 = \frac{y_1 - y_0}{x_1 - x_0} = \frac{\Delta y_0}{h}, \qquad a_2 = \frac{y_2 - 2y_1 + y_0}{(x_2 - x_0)(x_2 - x_1)} = \frac{\Delta^2 y_0}{2!\, h^2},$$

to jest

$$a_k = \frac{\Delta^k y_0}{k!\, h^k}, \qquad k = 0, 1, \dots, n.$$

Kad to uvrstimo u Newtonov oblik interpolacijskog polinoma

$$p_n(x) = a_0 + a_1(x - x_0) + a_2(x - x_0)(x - x_1) + \cdots + a_n(x - x_0)(x - x_1) \cdots (x - x_{n-1})$$

dobijemo prvi Newtonov interpolacijski polinom, to jest Newtonov interpolacijski polinom za interpolaciju unaprijed

$$N_n^I(x) = y_0 + \frac{\Delta y_0}{1!\, h}(x - x_0) + \frac{\Delta^2 y_0}{2!\, h^2}(x - x_0)(x - x_1) + \cdots + \frac{\Delta^n y_0}{n!\, h^n}(x - x_0) \cdots (x - x_{n-1}).$$

Da pojednostavimo izraz uvedemo član u,

$$u = \frac{x - x_0}{h},$$

pa je prvi Newtonov interpolacijski polinom sada oblika [6]

$$N_n^I(x) = y_0 + \frac{\Delta y_0}{1!} u + \frac{\Delta^2 y_0}{2!} u(u-1) + \cdots + \frac{\Delta^n y_0}{n!} u(u-1) \cdots (u - n + 1).$$

Greška interpolacije funkcije $f(x)$ koju radimo prvim Newtonovim interpolacijskim polinomom $N_n^I(x)$ glasi

$$R_n^I \leq \frac{|\Delta^{n+1} y|}{(n+1)!} |u(u-1) \cdots (u-n)|,$$

pri čemu se za $\Delta^{n+1} y$ uzima najveća vrijednost po svim apsolutnim vrijednostima konačne razlike $(n+1)$-og reda u odgovarajućim čvorovima, $\max_k |\Delta^{n+1} y_k|$. [15]

4.4.4.2. Drugi Newtonov interpolacijski polinom

U ovom se slučaju traži interpolacijski polinom od čvora x_n preko Newtonovog interpolacijskog polinoma koji je u obliku

$$N_n^{II}(x) = b_0 + b_1(x - x_n) + b_2(x - x_n)(x - x_{n-1}) + \cdots + b_n(x - x_n) \dots (x - x_1).$$

Koeficijente b_k odredimo iz svojstva $N_n^{II}(x_k) = y_k,\ k = 0, 1, \dots, n$. Zamjenom $x = x_n$, $x = x_{n-1}, \dots,\ x = x_0$ u gornju formulu $N_n^{II}(x)$ dobije se

$$b_0 = y_n, \qquad b_1 = \frac{y_n - y_{n-1}}{x_n - x_{n-1}} = \frac{\Delta y_{n-1}}{h}$$

to jest

$$b_k = \frac{\Delta y_{n-k}}{k!\, h^k}, \qquad k = 0, 1, \dots, n.$$

Ako uvedemo član v radi pojednostavljenja izraza,

$$v = \frac{x - x_n}{h},$$

dobivamo drugi Newtonov interpolacijski polinom ili Newtonov interpolacijski polinom s konačnim razlikama unazad [6]

$$N_n^{II}(x) = y_n + \frac{\Delta y_{n-1}}{1!} v + \frac{\Delta^2 y_{n-2}}{2!} v(v+1) + \cdots + \frac{\Delta^n y_0}{n!} v(v+1) \cdots (v+n-1).$$

Greška interpolacije funkcije $f(x)$ koju radimo drugim Newtonovim interpolacijskim polinomom $N_n^{II}(x)$ glasi [15]

$$R_n^{II} \leq \frac{|\Delta^{n+1} y|}{(n+1)!} |v(v+1) \cdots (v+n)|.$$

Zbog načina formiranja prvog i drugog Newtonovog interpolacijskog polinoma, svaki od njih pogodan je za interpolaciju čvorova na određenom dijelu intervala $[x_0, x_n]$. Prvi Newtonov interpolacijski polinom pogodan je za izvrednjavanje, to jest za procjenjivanje vrijednosti čvorova koji se nalaze u prvoj polovici intervala $[x_0, x_n]$, odnosno u okolini točke x_0, a drugi Newtonov interpolacijski polinom koristimo za interpolaciju čvorova koji se nalaze u drugoj polovici intervala $[x_0, x_n]$, to jest u okolini točke x_n.

Zadatak: Zadana je tablica funkcijom $f(x)$:

Tablica 9: Zadatak- ekvidistantni čvorovi

x	10	15	20	25	30	35	40	45	50
$f(x)$	0.2576	0.3415	0.4222	0.4991	0.5716	0.6392	0.7015	0.7573	0.8084

Potrebno je izračunati $f(13)$ i $f(48)$.

Rješenje:

Da bi riješili zadatak, najprije moramo formirati tablicu konačnih razlika:

Tablica 10: Zadatak-ekvidistantni čvorovi: tablica konačnih razlika

x	y	Δy	$\Delta^2 y$	$\Delta^3 y$	$\Delta^4 y$
10	0.2576				
		0.0839			
15	0.3415		-0.0032		
		0.0807		-0.0006	
20	0.4222		-0.0038		0
		0.0769		-0.0006	
25	0.4991		-0.0044		0.0001
		0.0725		-0.0005	
30	0.5716		-0.0049		0.0001
		0.0676		-0.0004	
35	0.6392		-0.0053		0.0002
		0.0623		-0.0002	
40	0.7015		-0.0055		0
		0.0558		-0.0002	
45	0.7573		-0.0057		
		0.0511			
50	0.8084				

Točka 13 nalazi se u prvoj polovici zadanog intervala $[10,50]$ pa ćemo za izračun vrijednosti $f(13)$ koristiti prvi Newtonov interpolacijski polinom:

$$N_3^I(x) = 0.2576 + 0.0839u - 0.0032\frac{u(u-1)}{2} - 0.0006\frac{u(u-1)(u-2)}{6}$$

gdje je

$$u = \frac{x - x_0}{h}, \qquad h = 5, \qquad x_0 = 10, \qquad x = 13,$$

dakle,

$$u = \frac{13 - 10}{5} = 0.6.$$

Temeljem toga približna vrijednost funkcije $f(x)$ u točki 13 glasi

$$f(13) \approx N_3^I(13) = 0.30829.$$

Ocjena greške iznosi

$$R_3^I \leq \frac{|\Delta^4 y|}{4!}|u(u-1)(u-2)(u-3)| = \frac{0.0002}{24} 0.6 \cdot 0.4 \cdot 1.4 \cdot 2.4 = 6.7 \cdot 10^{-6}.$$

Pošto je točka 48 na kraju intervala $[10,50]$ za izračun vrijednosti $f(48)$ koristit ćemo drugi Newtonov interpolacijski polinom:

$$N_3^{II}(x) = 0.8084 + 0.0511v - 0.0057\frac{v(v+1)}{2} - 0.0002\frac{v(v+1)(v+2)}{6}$$

gdje je

$$v = \frac{x - x_s}{h}, \qquad h = 5, \qquad x_8 = 50, \qquad x = 48,$$

pa je

$$v = \frac{48-50}{5} = -0.4.$$

$$f(48) \approx N_3^{II}(48) = 0.788657.$$

Ocjena greške iznosi

$$R_3^{II} \leq \frac{|\Delta^4 y|}{4!}|v(v+1)(v+2)(v+3)| = \frac{0.0002}{24} 0.4 \cdot 0.6 \cdot 1.6 \cdot 2.6 = 8.3 \cdot 10^{-6}.$$

5. Numeričko deriviranje

Problem koji će se ovdje rješavati je kako aproksimirati derivaciju funkcije f, koja je derivabilna, u nekoj točki x_0 i njezinim susjednim točkama $x_1, \dots, x_n$ koristeći samo vrijednosti funkcije f u zadanim točkama. Dakle, numeričko deriviranje možemo objasniti kao postupak približnog izračunavanja derivacije funkcije zadane tablicom $(x_i, f(x_i))$, pošto nam funkcija f nije poznata analitički. Isto tako, numeričko deriviranje možemo koristiti ako nam je analitički izraz funkcije f previše kompleksan. Svojstvo numeričkog deriviranja je to da je manje točno od interpolacije. [22] [23]

Niz tabličnih podataka može se derivirati: [24]

i) Deriviranjem interpolacijskih funkcija (polinoma), kao i

ii) Metodama za deriviranje (razne formule za deriviranje koje su izvedene iz Taylorovog polinoma, Newtonovog interpolacijskog polinoma, konačnih razlika,…)

5.1. Deriviranje interpolacijskih funkcija (polinoma)

Po definiciji je

$$f'(x) = \lim_{h \to 0} \frac{f(x+h) - f(x)}{h}.$$

Ako su točke x_i, x_{i+1} dovoljno blizu, to jest $h = x_{i+1} - x_i$ dovoljno mali, onda je

$$f'(x_i) \approx \frac{f(x_{i+1}) - f(x_i)}{h}, \qquad i = 0, 1, \dots, n-1.$$

Gornji izraz predstavlja derivaciju interpolacijskog polinoma 1. stupnja, provučenog kroz čvorove x_i i x_{i+1}. Da bi formula što bolje aproksimirala prvu derivaciju funkcije, razlika između čvorova x_i i x_{i+1}, to jest h, bi trebala biti čim manja, uz pretpostavku da je funkcija f dovoljno glatka. Međutim, to većinom vrijedi samo u teoriji pošto u praksi izmjereni podaci imaju neku pogrešku, u najmanju ruku zbog greške koja se javlja prilikom zaokruživanja. [5]

Ako je interpolacijski polinom p_n stupnja većeg od 1 i ako funkciju f aproksimiramo tim polinomom na segmentu $[a, b]$ gdje je polinom p_n

$$p_n(x) = a_0 + a_1 x + a_2 x^2 + \cdots + a_n x^n,$$

tada za $x \in [a, b]$ vrijedi

$$f'(x) \approx p_n'(x).$$

Postupak je isti i kod određivanja derivacija višeg reda [24]

$$f'(x) \approx p_n'(x) = a_1 + 2a_2 x + 3a_3 x^2 + \cdots + n\, a_n x^{n-1},$$

$$f''(x) \approx p_n''(x) = 2a_2 + 6a_3x + \cdots + n(n-1)a_nx^{n-2}.$$

Zadatak: Odredite vrijednost derivacije funkcije u točki $x = 4.2$ na temelju tablice.

Tablica 11: Zadatak-derivacija u točki $x = 4.2$

x	4.1	4.2	4.3
$f(x)$	0.243902	0.238095	0.232558

Najprije moramo odrediti interpolacijski polinom nekom od metoda koje smo prije objasnili pa tako imamo polinom drugog stupnja

$$p_2(x) = 0.5672x^2 - 4.76583x + 10.249173.$$

Nakon toga dobiveni polinom deriviramo i dobije se

$$p_2'(x) = 0.5672x - 4.76583.$$

Kad uvrstimo zadanu točku $x = 4.2$ u gornju jednadžbu dobivamo vrijednost prve derivacije u toj točki:

$$p_2'(4.2) = 0.5672 \cdot 4.2 - 4.76583 = -2.38359.$$

Ako znamo da su vrijednosti u tablici dobivene preko formule $f(x) = \frac{1}{x}$ možemo odrediti točnu vrijednost derivacije u zadanoj točki te tako usporediti dobivene rezultate kako bi vidjeli kolika je greška.

$$f'(4.2) = -\frac{1}{x^2} = -\frac{1}{4.2^2} = -0.056689$$

$$|f(4.2) - p_n(4.2)| = 0.000024.$$

Općenito, ako je poznata greška interpolacijskog polinoma e_n onda će biti poznata i pogreška derivacije e_n'. Dakle, imamo

$$f(x) = p_n(x) + e_n(x),$$

gdje je $p_n(x)$ interpolacijski polinom napisan, recimo, u Newtonovoj formi

$$p_n(x) = f[x_0] + (x - x_0)f[x_0, x_1] + \cdots + (x - x_0)\cdots(x - x_{n-1})f[x_0, x_1, \ldots, x_n],$$

a $e_n(x)$ greška interpolacijskog polinoma

$$e_n(x) = \frac{\omega(x)}{(n+1)!}f^{(n+1)}(\xi).$$

Ako deriviramo interpolacijski polinom, a zatim uvrstimo $x = x_0$ dobivamo

$$p_n'(x_0) = f[x_0, x_1] + (x_0 - x_1)f[x_0, x_1, x_2] + \cdots + (x_0 - x_1)\cdots(x_0 - x_{n-1})f[x_0, \ldots, x_n].$$

Ako pretpostavimo da funkcija f ima još jednu neprekidnu derivaciju, onda dobivamo i da je

$$e_n'(x_0) = \frac{f^{(n+1)}(\xi)}{(n+1)!}(x_0 - x_1)\cdots(x_0 - x_n).$$

Dakle, $p_n'(x_0)$ je aproksimacija derivacije funkcije f u točki x_0 i vrijedi [22]

$$f'(x_0) = p_n'(x_0) + e_n'(x_0).$$

Neki nedostatci numeričkog deriviranja interpolacijskog polinoma koje možemo izdvojiti su: mogućnost puno veće greške od same interpolacije, a ono može biti i računski zahtjevno. [24]

5.2. Metode za deriviranje

Ovdje ćemo se susresti s formulama za numeričko deriviranje koje su dobivene iz Taylorovog razvoja polinoma i formulama izvedenih iz konačnih razlika.

Kod korištenja Taylorove formule moramo najprije uvesti sljedeće oznake

$$y_i = f(x_i), \qquad y_i' = f'(x_i), \qquad h = x_{i+1} - x_i, \qquad i = 0, 1, \dots, n.$$

Ako u Taylorovu formulu prvog reda,

$$f(x) = f(x_0) + f'(x_0)(x - x_0)$$

uvrstimo $x = x_1$ dobivamo

$$y_1 = y_0 + y_0'h,$$

odnosno

$$y_0' = \frac{y_1 - y_0}{h}.$$

Jednako tako možemo, za $x = x_i$ napisati [5]

$$y_i' = \frac{y_{i+1} - y_i}{h}.$$

Formule za numeričko deriviranje mogu se izvesti i iz konačnih razlika unaprijed, ali i konačnih razlika unatrag. Rezultat koji se dobije isti je kao kod formula dobivenih Taylorovim razvojem. Dakle, imamo

$$f'(x_i) = \lim_{h \to 0} \frac{f(x_i + h) - f(x_i)}{h},$$

i ako uzmemo približnu vrijednost gornjeg izraza dobivamo formulu za konačnu razliku prvog reda unaprijed, kojom se može aproksimirati prva derivacija funkcije f

$$f'(x_i) \approx \frac{f(x_i + h) - f(x_i)}{h}.$$

Na isti način može se koristiti i konačna razlika prvog reda unazad [24]

$$f'(x_i) \approx \frac{f(x_i) - f(x_i - h)}{h}.$$

Zadatak: Odredite vrijednost derivacije funkcije f u točki $x = 4.2$.

x	4.1	4.2	4.3
$f(x)$	0.243902	0.238095	0.232558

Konačna razlika prvog reda unaprijed:

$$f'(4.2) = \frac{f(4.3) - f(4.2)}{4.3 - 4.2} = \frac{0.232558 - 0.238095}{0.1} = -0.05537$$

Konačna razlika prvog reda unazad:

$$f'(4.2) = \frac{f(4.2) - f(4.1)}{4.2 - 4.1} = \frac{0.238095 - 0.243902}{0.1} = -0.05807$$

Ako znamo da su vrijednosti u tablici dobivene funkcijom $f(x) = \frac{1}{x}$ imamo

$$f'(4.2) = -\frac{1}{x^2} = -\frac{1}{4.2^2} = -0.056689$$

možemo vidjeti i grešku konačne razlike prvog reda unaprijed

$$|-0.056689 - (-0.05537)| = -0.001319,$$

kao i grešku konačne razlike prvog reda unazad

$$|-0.056689 - (-0.05807)| = 0.001381.$$

6. Hermiteova interpolacija

Lagrangeov interpolacijski polinom formirali smo na temelju podataka o točkama kroz koje krivulja treba prolaziti, a Hermiteov polinom formira se na temelju podataka o točkama kroz koje krivulja treba proći, ali i o podacima o vrijednostima derivacija funkcije u tim točkama. Dakle, osim vrijednosti funkcije f u čvorovima, zadane su i vrijednosti derivacije te funkcije u čvorovima. [25]

Sada ćemo objasniti egzistenciju i jedinstvenost Hermiteovog polinoma.

Teorem. *Postoji jedinstveni polinom* h_{2n+1} *stupnja najviše* $2n+1$, *koji zadovoljava interpolacijske uvjete*

$$h_{2n+1}(x_i) = f_i, \qquad h'_{2n+1}(x_i) = f_i', \qquad i = 0, \dots, n,$$

gdje su x_i *međusobno različite točke i* f_i, f_i' *zadani realni brojevi.*

Dokaz. Egzistenciju polinoma $h_{2n+1}(x)$ dokazat ćemo konstrukcijom baze, slično kao i kod Lagrangeovog oblika interpolacijskog polinoma. Neka su

$$h_{i,0}(x) = [1 - 2(x - x_i)\ell_i'(x_i)]\ell_i^2(x)$$

$$h_{i,1}(x) = (x - x_i)\ell_i^2(x),$$

gdje su ℓ_i funkcije Lagrangeove baze koje smo susreli kod Lagrangeovog oblika interpolacijskog polinoma

$$\ell_i(x) = \prod_{\substack{j=0 \\ j \neq i}}^{n} \frac{x - x_j}{x_i - x_j}, \qquad i = 0, \dots, n.$$

Možemo provjeriti da su $h_{i,0}(x)$ i $h_{i,1}(x)$ polinomi stupnja $2n+1$ koji zadovoljavaju sljedeće relacije

$$h_{i,0}(x_j) = \delta_{ij}, \qquad h_{i,1}(x_j) = 0,$$

$$h'_{i,0}(x_j) = 0, \qquad h'_{i,1}(x_j) = \delta_{ij}, \qquad i, j = 0, \dots, n.$$

gdje je δ_{ij} Kroneckerov simbol. Ako se polinom definira kao

$$h_{2n+1}(x) = \sum_{i=0}^{n} \left(f_i h_{i,0}(x) + f_i' h_{i,1}(x)\right),$$

lagano se provjeri da h_{2n+1} zadovoljava uvjete teorema. Jedinstvenost tog polinoma moramo posebno dokazati. Neka je $q_{2n+1}(x)$ polinom koji također ispunjava interpolacijske uvjete teorema. Tada je $h_{2n+1}(x) - q_{2n+1}(x)$ polinom stupnja ne većeg od $2n+1$, s dvostrukim nultočkama u svakom čvoru interpolacije x_i, odnosno barem $2n+2$ nultočke, što je moguće samo ako je taj polinom jednak nuli.

Polinomi $h_{i,0}$, $h_{i,1}$ zovu se Hermiteove baze, a polinom h_{2n+1} nazivamo Hermiteovim interpolacijskim polinomom. Ti polinomi označavaju se i s H_n, $n \in \mathbb{N}_0$. [22]

Hermiteovi polinomi zadovoljavaju rekurzivnu relaciju

$$H_{n+1}(x) - 2xH_n(x) + 2nH_{n-1}(x) = 0$$

uz početne uvjete $H_0(x) = 1,\ H_1(x) = 2x$. [26]

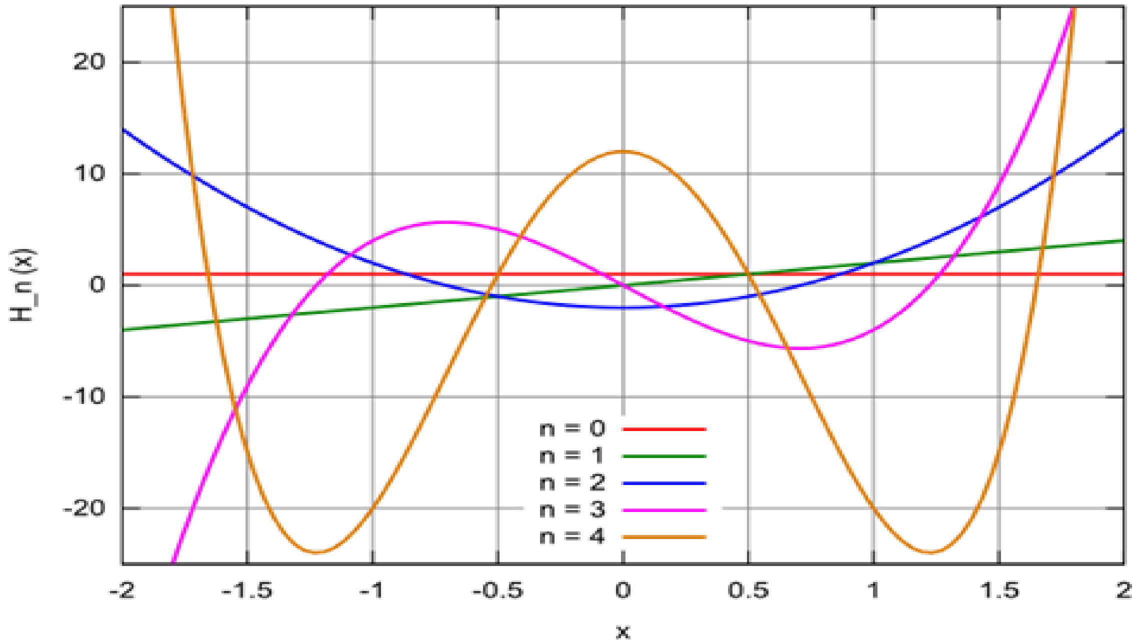
Prvih nekoliko Hermiteovih polinoma:

$$H_0(x) = 1,$$
$$H_1(x) = 2x,$$
$$H_2(x) = 4x^2 - 2,$$
$$H_3(x) = 8x^3 - 12x,$$
$$H_4(x) = 16x^4 - 48x^2 + 12,$$
$$H_5(x) = 32x^5 - 160x^3 + 120x,$$
$$H_6(x) = 64x^6 - 480x^4 + 720x^2 - 120.$$

n-ti Hermiteov polinom H_n je rješenje diferencijalne jednadžbe [26]

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

Sljedeća slika prikazuje nekoliko prvih Hermiteovih polinoma.



Slika 4:Prvih nekoliko Hermiteovih polinoma

Izvor:[*www.mathepedia.de/html/a_analysis/8_dim1/f_spezfunc/c_spezpolynome/Hermite.aspx?w=500&h=400*]

Hermiteov interpolacijski polinom može se napisati i u Newtonovoj bazi. Točke interpolacije biti će $x_0, x_0, x_1, x_1, \dots, x_n, x_n$. Vidimo da je svaki čvor dvostruki. Neka su x_0 i $x_1 = x_0 + h$ dva čvora i pustimo da $h \to 0$. Tada imamo

$$f[x_0, x_0] = \lim_{h \to 0} f[x_0, x_0 + h]$$
$$= \lim_{h \to 0} \frac{f(x_0 + h) - f(x_0)}{h} = f'(x_0).$$

Za ponovljene čvorove podijeljene razlike se definiraju kao

$$f[x_i, x_i] \coloneqq f'(x_i), \qquad f[x_i, x_i, x_i] \coloneqq \frac{f''(x_i)}{2!}, \qquad f[x_i, \dots, x_i] \coloneqq \frac{f^{(n)}(x_i)}{n!}.$$

Ostale podijeljene razlike računaju se na uobičajeni način. To ćemo prikazati na sljedećoj tablici. [27]

Tablica 12: Tablica podijeljenih razlika za Hermiteovu interpolaciju

x_k	$f[x_k]$	$f[x_k, x_{k+1}]$	$f[x_k, x_{k+1}, x_{k+2}]$	$\cdots$	$f[x_0, \dots, x_n]$
x_0	$f[x_0]$				
		$f'(x_0)$			
x_0	$f[x_0]$		$f[x_0, x_0, x_1]$	$\ddots$	
		$f[x_0, x_1]$			
x_1	$f[x_1]$		$f[x_0, x_1, x_1]$		
		$f'(x_1)$			$\cdots$
x_1	$f[x_1]$		$f[x_1, x_1, x_2]$		
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\cdots$	
x_n	$f[x_n]$		$f[x_{n-1}, x_n, x_n]$		
		$f'(x_n)$			
x_n	$f[x_n]$				

Hermiteov interpolacijski polinom sada izgleda

$$h_{2n+1}(x) = f[x_0] + f'(x_0)(x - x_0) + f[x_0, x_0, x_0](x - x_0)^2 +$$
$$+ f[x_0, x_0, x_1, x_1](x - x_0)^2(x - x_1) + \dots +$$
$$+ f[x_0, x_0, \dots, x_n, x_n](x - x_0)^2 \cdots (x - x_{n-1})^2(x - x_n).$$

Zadatak: Odredimo Hermiteov interpolacijski polinom za funkciju $f(x)$ zadanu tablicom: [28]

Tablica 13: Zadatak- Hermiteova interpolacija

x_i	$f(x_i)$	$f'(x_i)$	$f''(x_i)$
0	-1	-2	$-$
1	0	10	40

Iz tablice vidimo da je čvor $x_0 = 0$ dvostruki, a $x_1 = 1$ trostruki čvor, pa je ovaj polinom oblika

$$P_4(x) = f(x_0) + f[x_0, x_0](x - x_0) + f[x_0, x_0, x_1](x - x_0)^2 + f[x_0, x_0, x_1, x_1](x - x_0)^2(x - x_1) + f[x_0, x_0, x_1, x_1, x_1](x - x_0)^2(x - x_1)^2.$$

Koristeći definiciju podijeljenih razlika i svojstvo $f[x_i, \dots, x_i] \coloneqq \frac{f^{(n)}(x_i)}{n!}$ slijedi da je

$$f[x_0, x_0] = f'(x_0) = -2$$

$$f[x_0, x_0, x_1] = \frac{1}{x_1 - x_0}\left(\frac{f(x_1) - f(x_0)}{x_1 - x_0} - f'(x_0)\right) = 3$$

$$f[x_0, x_0, x_1, x_1] = \frac{1}{x_1 - x_0}\left(\frac{1}{x_1 - x_0}\left(f'(x_1) - \frac{f(x_1) - f(x_0)}{x_1 - x_0}\right) - f[x_0, x_0, x_1]\right) = 6$$

$$f[x_0, x_0, x_1, x_1, x_1] = \frac{1}{x_1 - x_0}\left(\frac{1}{x_1 - x_0}\left(\frac{f''(x_1)}{2} - \frac{f'(x_1)}{x_1 - x_0} + \frac{f(x_1) - f(x_0)}{(x_1 - x_0)^2}\right) - f[x_0, x_0, x_1, x_1]\right) = 5$$

pa tako imamo

$$P_4(x) = -1 - 2x + 3x^2 + 6x^2(x - 1) + 5x^2(x - 1)^2 = 5x^4 - 4x^3 + 2x^2 - 2x - 1.$$

6.1. Greška Hermiteove interpolacije

Pogrešku Hermiteove interpolacije možemo dobiti na sličan način kao i pogrešku Lagrangeove interpolacije, samo što moramo iskoristiti da je h_{2n+1} stupnja $2n + 1$, a oblik polinoma čvorova $\omega_n(x) = \omega^2(x)$. Tako pogreška kod interpolacije Hermiteovim polinomom h_{2n+1} funkcije $f \in C^{(2n+2)}[x_{min}, x_{max}]$ u $n + 1$ čvorova $x_0, \dots, x_n$ glasi

$$e(x) = f(x) - h_{2n+1}(x) = \frac{f^{(2n+2)}(\xi)}{(2n+2)!}\omega^2(x),$$

gdje su ω i ξ definirani isto kao i u pogrešci Lagrangeovim interpolacijskim polinomom. [27]

7. Optimalni izbor čvorova interpolacije

7.1. Izbor stupnja polinoma

Ako imamo ekvidistantne interpolacijske čvorove, za izbor stupnja polinoma koristi se kriterij koji slijedi iz međusobne veze derivacija polinoma i konačnih razlika. Funkcija se ponaša kao polinom m-tog stupnja ukoliko su konačne razlike m-tog reda približno konstantne, to jest razlike $(m+1)$-og reda približno jednake nuli. Zbog tog se razloga za stupanj interpolacijskog polinoma bira m. Analogan kriterij konstantnih podijeljenih razlika primjenjuje se kada čvorovi nisu ekvidistantni. Definiranje i primjena toga kriterija nisu uključeni u ovaj rad. $[1]$

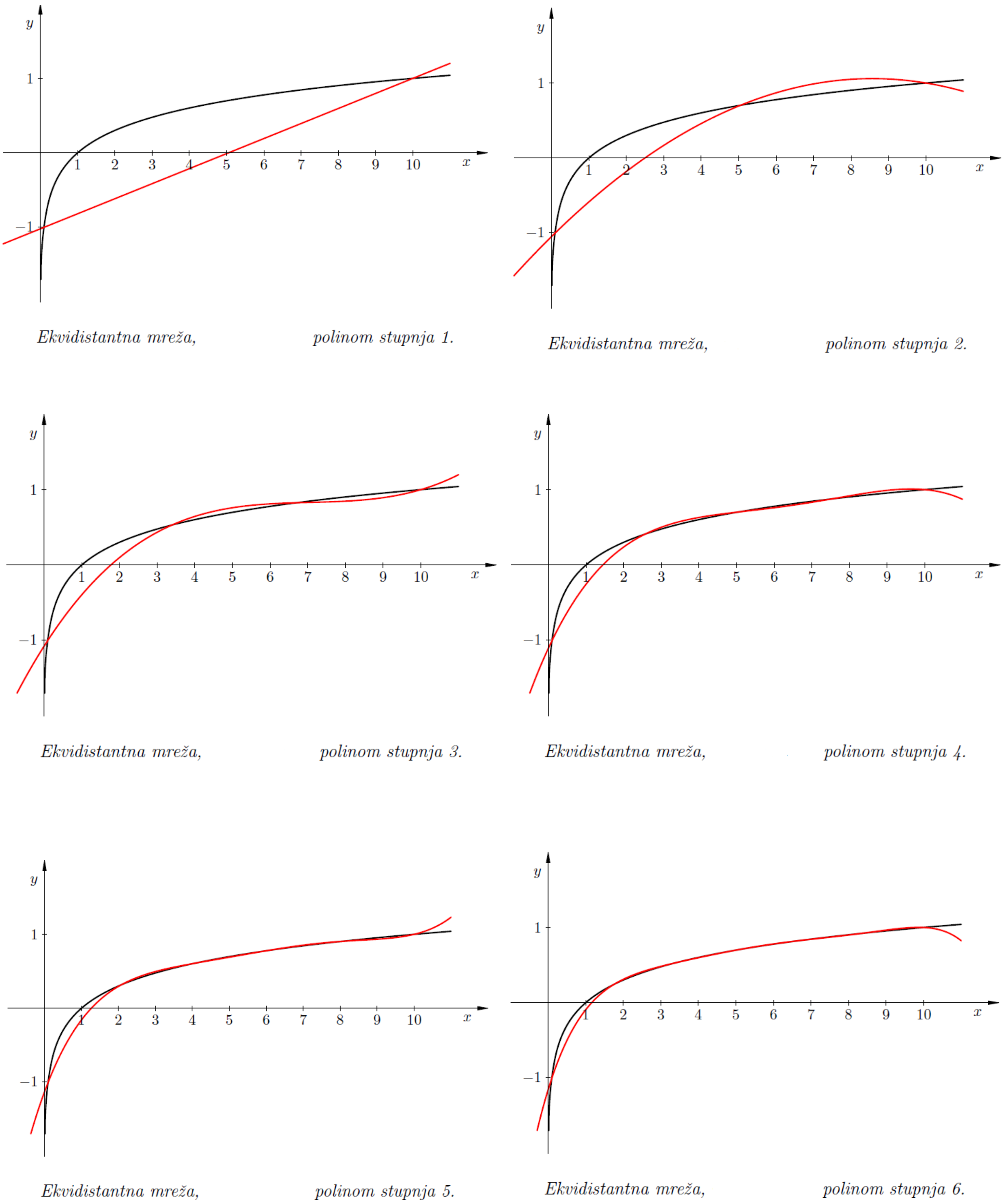
7.2. Izbor čvorova interpolacije

Pošto je za stupanj interpolacijskog polinoma odabran m, pitamo se kako izabrati skup od $(m+1)$ susjednih interpolacijskih čvorova $x_k, x_{k+1}, \dots, x_{k+m}$, $(0 \leq k \leq n-m)$. Apsolutna vrijednost produkta koji je definiran čvorovima interpolacije i koji figurira u izrazu za grešku interpolacije,

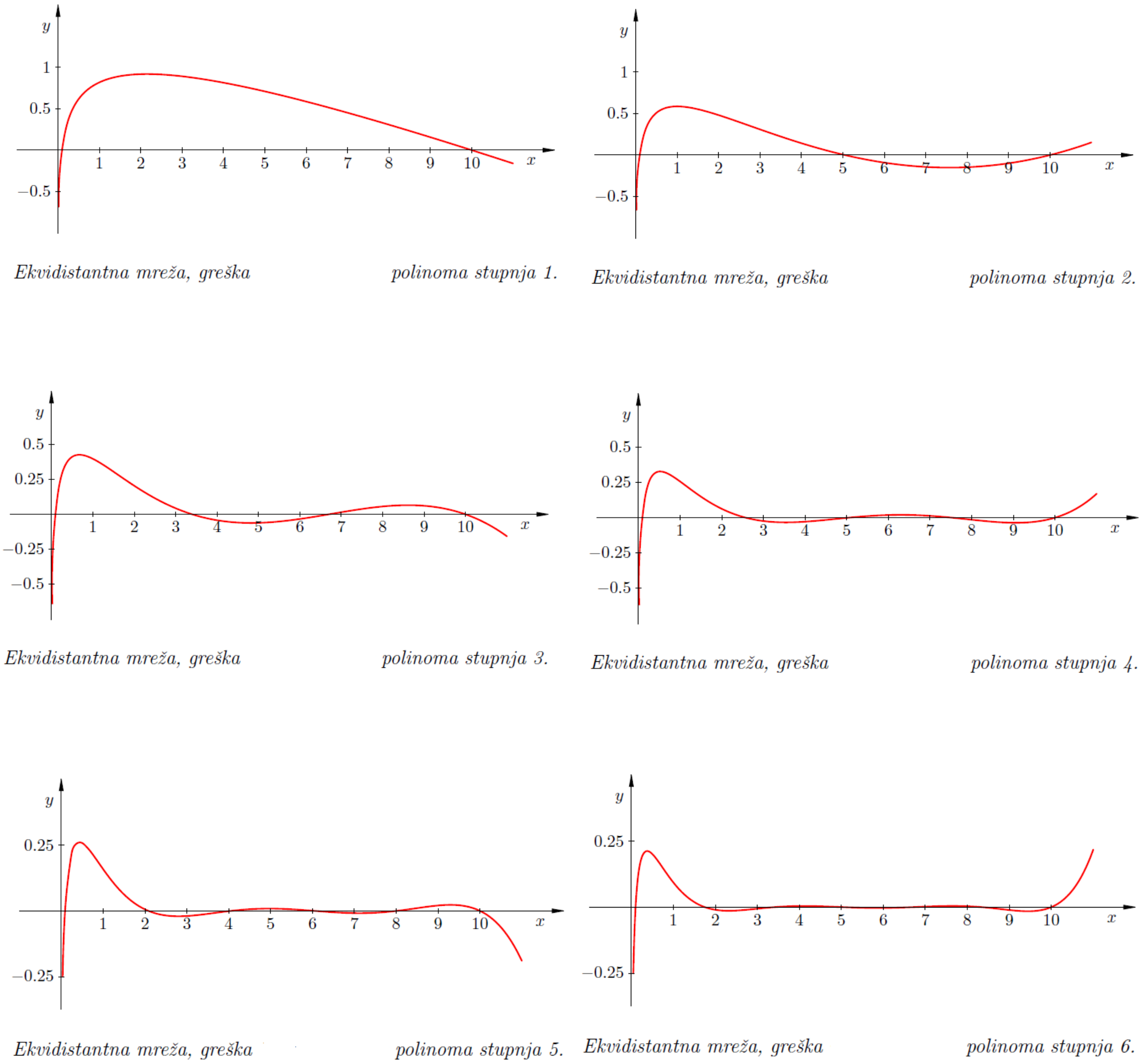
$$\prod\nolimits_{m+1}(x) = \prod_{i=k}^{k+m}(x - x_i),$$

ima minimum u sredini intervala $[x_k, x_{k+m}]$. Zato se interpolacijski čvorovi biraju tako da točka x, u kojoj se računa vrijednost funkcije bude što bliža sredini intervala $[x_k, x_{k+m}]$. $[1]$

Prvi koji je uočio probleme koji nastaju kod interpolacije s ekvidistantnim čvorovima, odnosno interpolacije na ekvidistantnoj mreži bio je njemački matematičar Runge. On je konstruirao funkciju sa svojstvom da niz Newtonovih interpolacijskih polinoma na ekvidistantnoj mreži ne konvergira prema toj funkciji. Ovdje ćemo prikazati grafove interpolacijskih polinoma stupnjeva $1-6$ koji interpoliraju funkciju $f(x) = \log(x)$ za $x \in [0.1, 10]$ na ekvidistantnoj mreži kako bi vidjeli slična svojstva koja je uočio i Runge. Prvo ćemo prikazati interpolacijske polinome na ekvidistantnoj mreži, a zatim greške tih polinoma. Funkcija je prikazana crnom bojom, a interpolacijski polinomi crvenom. $[29]$



Slika 5: Interpolacijski polinomi stupnja 1-6 na ekvidistantnoj mreži
Izvor: [*29*]



Slika 6: Greške interpolacijskih polinoma stupnja 1-6 na ekvidistantnoj mreži
Izvor: [*29*]

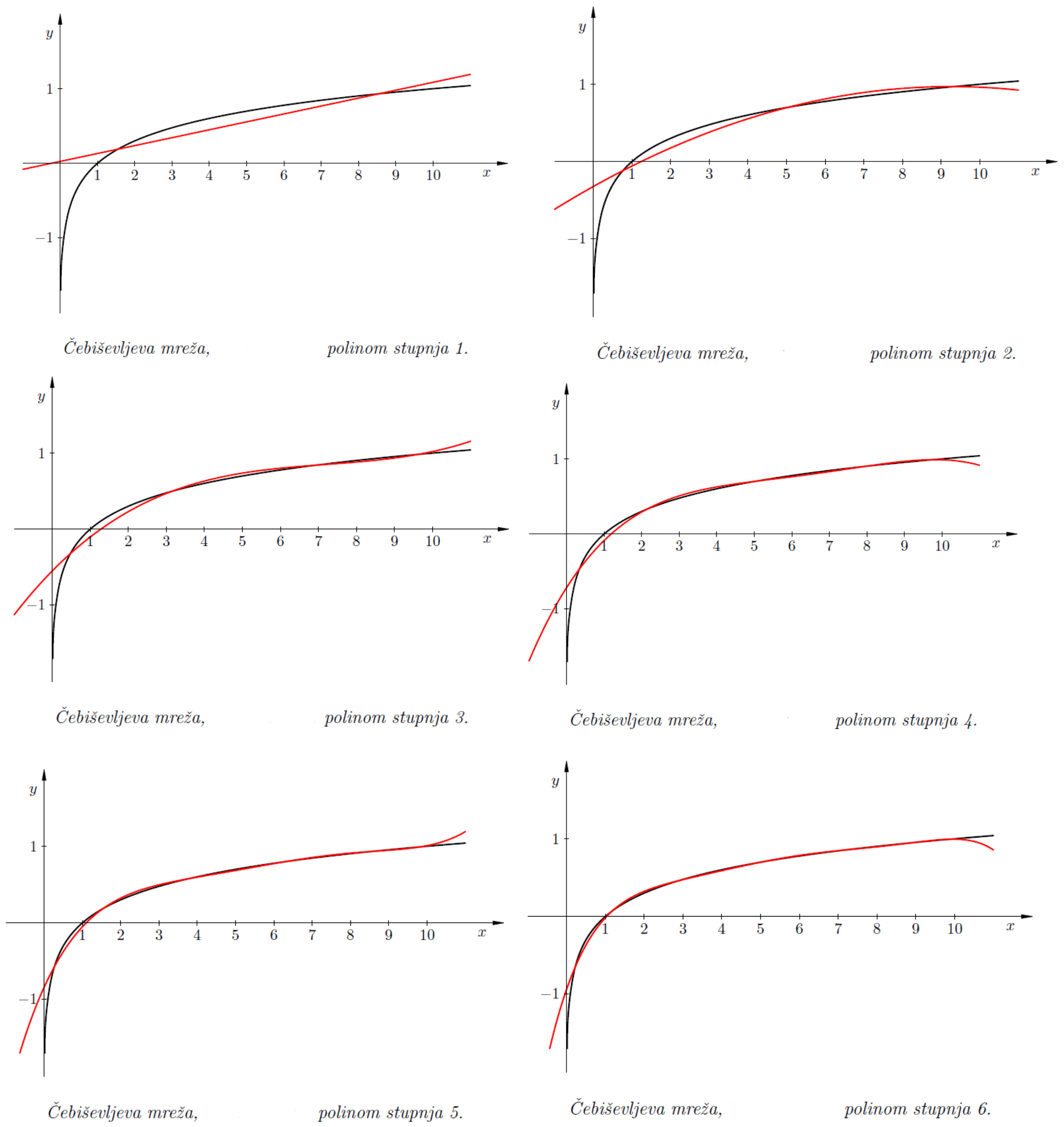
Ako se u tom primjeru umjesto ekvidistantnih čvorova interpolacije koriste neekvidistantne, točnije tzv. Čebiševljeve točke na intervalu $[a, b]$,

$$x_k = \frac{1}{2}\left(a + b + (a - b)\cos\frac{2k+1}{2n+2}\right), \qquad k = 0, \ldots, n$$

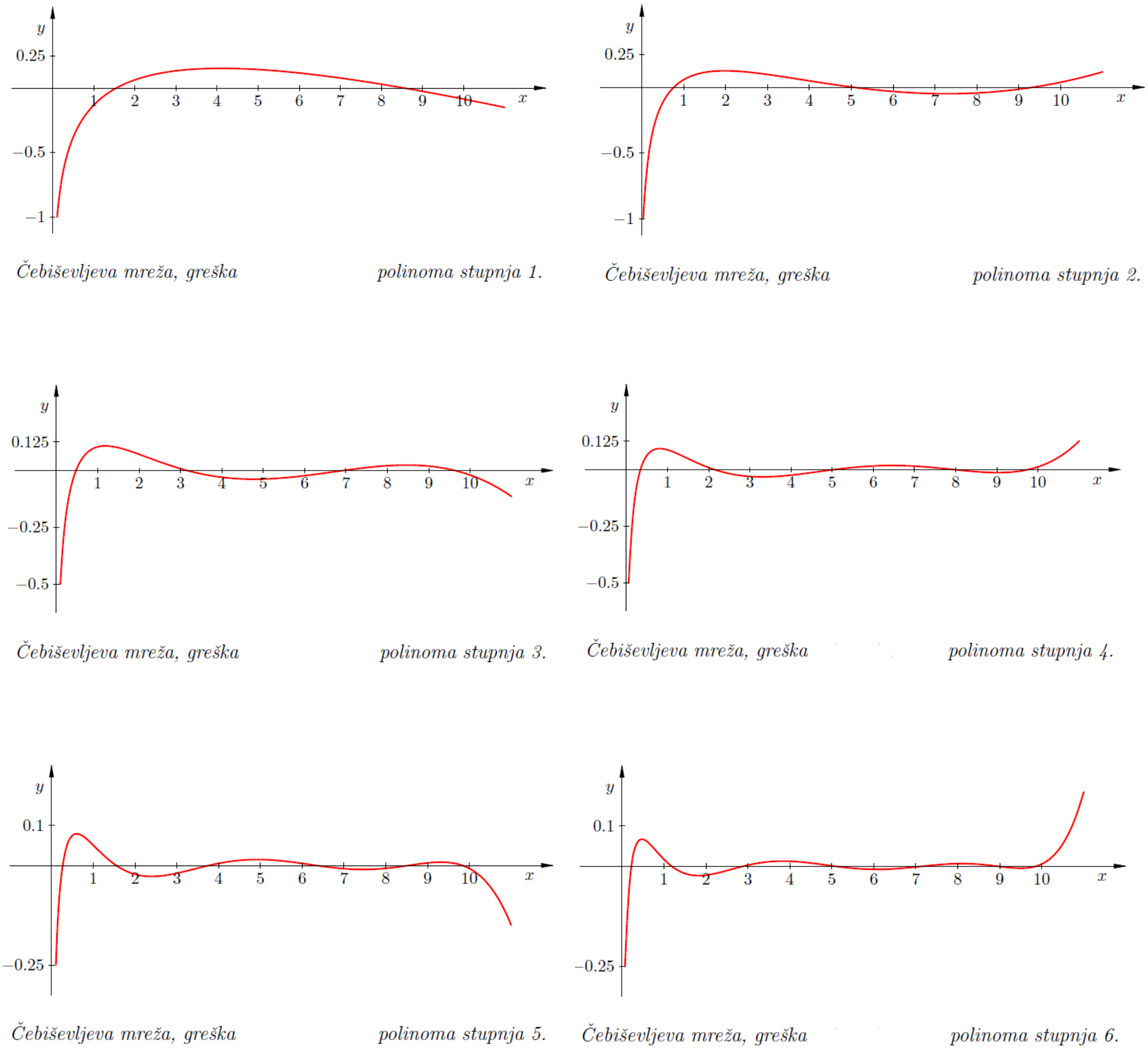
onda će rezultat biti bolji, a greške manje. Uzlazno poredane Čebiševljeve točke na intervalu $[a, b]$ su oblika [29]

$$x_k = \frac{1}{2}\left(a + b + (a - b)\cos\frac{(2k+1)\pi}{2n+2}\right), \qquad k = 0, \ldots, n.$$

Ovdje ćemo prikazati kako je funkcija $f(x) = \log(x)$ za $x \in [0.1, 10]$ interpolirana Čebiševljenim polinomima stupnja $1 - 6$, a prikazat ćemo i njihove greške. Isto kao i prije, funkcija je crne boje, a interpolacijski polinom crvene.



Slika 7: Interpolacijski polinomi stupnja 1-6 na Čebiševljevoj mreži
Izvor: [*29*]



Slika 8: Greške interpolacijskih polinoma stupnja 1-6 na Čebiševljevoj mreži
Izvor: [*29*]

Vidimo da Čebiševljeve točke bolje aproksimiraju zadanu funkciju od ekvidistantnih čvorova s povećanjem stupnja, a i greške tih polinoma su puno manje. Na slikama su grafovi grešaka skoro pa identični, ali skala na y osi kod Čebiševljeve mreže puno je manja. Pokažimo zašto su Čebiševljeve točke bolji izbor za minimizaciju pogreške, odnosno optimalni izbor čvorova interpolacije od ekvidistantnih čvorova. Znamo da greška interpolacijskog polinoma stupnja n iznosi

$$f(x) - p_n(x) = \frac{(x - x_0) \cdots (x - x_n)}{(n+1)!} f^{(n+1)}(\xi).$$

Ideja koja se koristi za optimalni izbor čvorova interpolacije je minimizacija faktora koji ovisi samo o čvorovima, a to je

$$(x - x_0) \cdots (x - x_n).$$

Minimizacija se provodi globalno, preko cijelog intervala $[-1, 1]$, i to uniformno – u max formi. Točke interpolacije $x_j \in [-1, 1]$ biraju se tako da minimiziraju

$$\max_{-1 \leq x \leq 1} |(x - x_0) \cdots (x - x_n)| \ (\Delta)$$

i tako dobivamo problem minimaks aproksimacije. Ovdje koristimo Čebiševljeve polinome (prve vrste) i jedno njihovo svojstvo, a to je

$$M(x) = \frac{1}{2^{n-1}} T_n(x).$$

Minimum u jednadžbi (Δ) dobit ćemo ako stavimo

$$(x - x_0) \cdots (x - x_n) = \frac{1}{2^n} T_{n+1}(x),$$

a minimalna vrijednost bit će $1/2^n$. Odatle čitamo da su čvorovi $x_0, \dots, x_n$ nultočke polinoma T_{n+1}, i glase, kao što smo već prije naveli, u silaznom poretku,

$$x_k = \cos\left(\frac{(2k+1)\pi}{2n+2}\right), \qquad k = 0, \dots, n.$$

Uzlazni poredak čvorova glasi [29]

$$x_k = \cos\left(\frac{(2(n-k)+1)\pi}{2n+2}\right), \qquad k = 0, \dots, n.$$

Na nekom intervalu $[a, b]$ uzlazno poredane Čebiševljeve točke su [18]

$$x_k = \frac{a+b}{2} + \frac{b-a}{2} * \cos\frac{(2(n-k)+1)\pi}{2n+2}, \qquad k = 0, \dots, n.$$

Sada ćemo na primjeru pokazati interpolacijski polinom na ekvidistantnoj i Čebiševljevoj mreži, kao i ocjenu greške polinoma. Za funkciju

$$f(x) = \frac{1}{1 + 25x^2}$$

na intervalu $[-1, 1]$ potrebno je pronaći interpolacijski polinom stupnja 2 na ekvidistantnoj i Čebiševljevoj mreži. Treba napraviti i uniformnu ocjenu greške. [13]

Rješenje:

Ekvidistantna mreža

$$x_i = x_0 + ih, \qquad h = \frac{x_2 - x_0}{2} = \frac{1 - (-1)}{2} = 1,$$

$$x_0 = -1, \qquad x_1 = 0, \qquad x_2 = 1.$$

Ekvidistantna mreža sastoji se od čvorova -1, 0 i 1. Tablica podijeljenih razlika za ekvidistantne čvorove izgleda ovako

Tablica 14: Zadatak- ekvidistantna i Čebiševljeva mreža, tablica podijeljenih razlika

x_i \ k	0	1	2
-1	$\mathbf{1/26}$		
		$\mathbf{25/26}$	
0	1		$\mathbf{-25/26}$
		$-25/26$	
1	$1/26$		

Interpolacijski polinom stupnja 2 tada je oblika

$$p_2(x) = \frac{1}{26} + \frac{25}{26}(x+1) - \frac{25}{26}(x+1)x = 1 - \frac{25}{26}x^2.$$

Čebiševljeva mreža

Kod Čebiševljeve mreže čvorovi su dani kao nultočke polinoma T_{n+1}, to jest $T_3(x) = \cos(3\arccos x)$, a za njih vrijedi

$$x_i = \cos\frac{(2i+1)\pi}{6}, \qquad i = 0,1,2$$

pa tako imamo

$$x_0 = \cos\left(\frac{\pi}{6}\right) = \frac{\sqrt{3}}{2}, \qquad x_1 = \cos\left(\frac{3\pi}{6}\right) = 0, \qquad x_2 = \cos\left(\frac{5\pi}{6}\right) = -\frac{\sqrt{3}}{2}.$$

Tablica podijeljenih razlika za Čebiševljeve čvorove izgleda ovako.

Tablica 15: Zadatak- ekvidistantna i Čebiševljeva mreža, tablica podijeljenih razlika 2

x_i \ k	0	1	2
$\frac{\sqrt{3}}{2}$	$\frac{4}{79}$		
		$-\frac{150}{79\sqrt{3}}$	
0	1		$-\frac{100}{79}$
		$\frac{150}{79\sqrt{3}}$	
$-\frac{\sqrt{3}}{2}$	$\frac{4}{79}$		

Interpolacijski polinom kod Čebiševljeve mreže tako je oblika

$$r_2(x) = \frac{4}{79} - \frac{150}{79\sqrt{3}}\left(x - \frac{\sqrt{3}}{2}\right) - \frac{100}{79}\left(x - \frac{\sqrt{3}}{2}\right)x = 1 - \frac{100}{79}x^2.$$

Za ocjenu pogreške potreban nam je maksimum $\left|f^{(3)}(x)\right|$ na intervalu $[-1, 1]$. Kako je

$$f'(x) = -\frac{50x}{(1+25x^2)^2},$$

$$f''(x) = -\frac{50(1-75x^2)}{(1+25x^2)^3},$$

$$f'''(x) = -15000\frac{25x^3 - x}{(1+25x^2)^4},$$

$$f^{(4)}(x) = \frac{15000}{(1+25x^2)^5}(3125x^4 - 250x^2 + 1),$$

trebamo pronaći rješenje jednadžbe

$$3125x^4 - 250x^2 + 1 = 0 \iff 3125u^2 - 250u + 1 = 0$$

i dobijemo

$$u_1 = \frac{5 + 2\sqrt{5}}{125} = 0.075777, \qquad u_2 = \frac{5 - 2\sqrt{5}}{125} = 0.004922$$

pa su nultočke četvrte derivacije

$$x_{1,2} = \pm 0.275, \qquad x_{3,4} = \pm 0.06498.$$

Da bi se pronašao maksimum $\left|f^{(3)}(x)\right|$ na zadanom intervalu $[-1, 1]$, potrebno je pogledati vrijednosti od $\left|f^{(3)}(x)\right|$ u točkama ekstrema i u rubovima, a kako je $f^{(3)}(x)$ neparna funkcija, dovoljno je pogledati samo za pozitivne točke. Tako imamo

$$\left|f^{(3)}(x_1)\right| = 52.62, \qquad \left|f^{(3)}(x_3)\right| = 583.57, \qquad \left|f^{(3)}(1)\right| = 0.787.$$

$$\max_{x\in[-1,1]}\left|f^{(3)}(x)\right| = 583.57.$$

U ovom je slučaju ocjena maksimuma neefikasna. Za uniformnu ocjenu greške kod ekvidistantne mreže potrebno je pronaći

$$\max_{x\in[-1,1]}|(x+1)x(x-1)|.$$

Sada promatramo polinom $s(x) = x^3 - x$. On postiže ekstreme u točkama $\pm 1/\sqrt{3}$ i po apsolutnoj vrijednosti oni iznose $2\sqrt{3}/9$. Uniformna ocjena greške u ekvidistantnom slučaju tako iznosi

$$\max_{x\in[-1,1]}|f(x) - p_2(x)| \leq \frac{2\sqrt{3}/9}{6} \cdot 583.57 = 37.436.$$

Za interpolacijski polinom na Čebiševljevoj mreži vrijedi

$$\max_{x\in[-1,1]}\left(x+\frac{\sqrt{3}}{2}\right)x\left(x-\frac{\sqrt{3}}{2}\right)=2\cdot\left(\frac{1-(-1)}{4}\right)^{2+1}=\frac{1}{4},$$

pa uniformna ocjena pogreške iznosi

$$\max_{x\in[-1,1]}|f(x)-r_2(x)|\leq\frac{1/4}{6}\cdot 583.57=24.315.$$

Možemo još napomenuti da prava pogreška u slučaju ekvidistantne mreže iznosi 0.646229, a kod Čebiševljeve mreže 0.6005977.

8. Čebiševljevi polinomi

U ovom ćemo se poglavlju upoznati s Čebiševljevim polinomima prve i druge vrste, kao i o njihovim raznim svojstvima. Čebiševljevi polinomi prve vrste i druge vrste, isto kao i Hermiteovi polinomi pripadaju skupu ortogonalnih polinoma. Dva su polinoma ortogonalna ako je njihov skalarni produkt jednak nuli. Čebiševljevi polinomi, jednako kao i Hermiteovi polinomi, nastali su rješavanjem diferencijalnih jednadžbi drugog reda. [26]

8.1. Čebiševljevi polinomi prve vrste

Čebiševljevi polinomi prve vrste označavaju se s $T_n,\ n \in \mathbb{N}_0$. Čebiševljev polinom prve vrste n-tog stupnja T_n, jedno je od rješenja diferencijalne jednadžbe

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

Ti polinomi definirani su sa

$$T_n(\cos\theta) = \cos(n\theta).$$

Ako uvedemo supstituciju $x = \cos\theta$ dobije se

$$T_n(x) = \cos(n \arccos(x)),$$

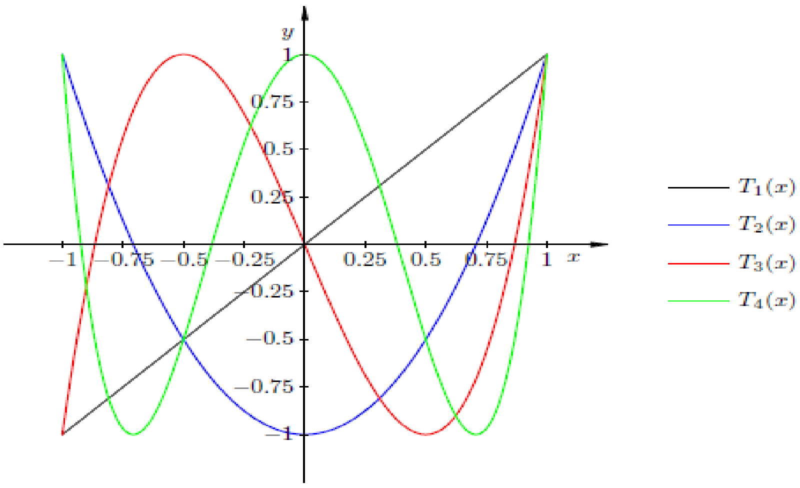
gdje je $x \in [-1, 1]$, što nam daje formulu za računanje Čebiševljevih polinoma prve vrste. Tako imamo nekoliko prvih Čebiševljevih polinoma prve vrste [26]

$$T_0 = \cos 0 = 1,$$

$$T_1 = \cos(arc\cos x) = x,$$

$$T_2 = 2cos^2(arc\cos x) - 1 = 2x^2 - 1.$$

Sljedeća slika prikazuje graf nekoliko prvih Čebiševljevih polinoma.



Slika 9: Čebiševljevi polinomi prve vrste
Izvor: [*18*]

Budući da vrijedi

$$\cos(n+1)\,\alpha + \cos(n-1)\,\alpha = \cos(n\alpha + \alpha) + \cos(n\alpha - \alpha)$$
$$= \cos n\alpha \cos\alpha - \sin n\alpha \sin\alpha + \cos n\alpha \cos\alpha + \sin n \sin\alpha$$
$$= 2\cos n\alpha \cos\alpha$$

ako stavimo $\alpha = \arccos x$, dobivamo

$$\cos((n+1)\arccos x) + \cos((n-1)\arccos x) = 2\cos(n\arccos x)\cos(\arccos x).$$

Pomoću gornje jednadžbe dobivamo rekurzivnu relaciju Čebiševljevih polinoma prve vrste koja glasi

$$T_{n+1}(x) - 2xT_n(x) + T_{n-1}(x) = 0,$$

odnosno

$$T_n(x) = 2x\,T_{n-1}(x) - T_{n-2}(x),$$

uz uvjete $T_0(x) = 1$ i $T_1(x) = x$.

Odatle slijedi npr. [2]

$$T_2(x) = 2x^2 - 1,$$
$$T_3(x) = 4x^3 - 3x,$$
$$T_4(x) = 8x^4 - 8x^2 + 1,$$
$$T_5(x) = 16x^5 - 20x^3 + 5x,$$
$$\vdots$$
$$T_{10}(x) = 512x^{10} - 1280x^8 + 1120x^6 - 400x^4 + 50x^2 - 1.$$

Kao što smo već vidjeli kod optimalnog izbora čvorova interpolacije, na nekom intervalu $[a,b]$ uzlazno poredane Čebiševljeve točke su [18]

$$x_k = \frac{a+b}{2} + \frac{b-a}{2} * \cos\frac{(2(n-k)+1)\pi}{2n+2}, \qquad k = 0, \dots, n.$$

Ako promatramo interval $[-1,1]$, postoji n različitih nultočaka koje su dane formulom

$$x_k = cos\frac{(2k-1)\pi}{2n}, \qquad k = 1,2,\dots,n.$$

Čebiševljev polinom $T_n(x)$ za $x \in [-1,1]$ ima $n+1$ različitu točku

$$x_k = \cos\frac{k\pi}{n}, \qquad k = 0,1,\dots,n$$

u kojima naizmjenično postiže globalne maksimume i minimume, odnosno svoje ekstreme. [2]

Ovdje ćemo dokazati svojstvo koje smo koristili kod optimalnog izbora čvorova interpolacije. [30]

Teorem. *Polinom* $\frac{1}{2^{n-1}}T_n(x)$ *ima svojstvo da među svim polinomima stupnja* n *s vodećim koeficijentom* 1 *najmanje odstupa od nule na intervalu* $[-1\,,1]$. *Maksimalno odstupanje tog polinoma je* $\pm\frac{1}{2^{n-1}}$ *i ono se dostiže u* $n+1$ *točki.*

Dokaz. Neka je $P_n(x) = x^n +$ bilo koji polinom s vodećim koeficijentom 1 i neka vrijedi

$$P_n(x) \leq \frac{1}{2^{n-1}},$$

gdje je $|x| \leq 1$. Potrebno je dokazati da je

$$P_n(x) = \frac{1}{2^{n-1}}T_n(x).$$

U dokazivanju toga koristit ćemo svojstvo Čebiševljevih polinoma koje glasi: ako je

$$x_k = \cos\left(\frac{k\pi}{n}\right),$$

tada vrijedi

$$T_n(x_k) = \cos k\pi = (-1)^k, \qquad k = 0, 1, \dots, n.$$

Pogledajmo sada polinom

$$Q(x) = \frac{1}{2^{n-1}}T_n(x) - P_n(x).$$

Stupanj polinoma $Q(x)$ ne prelazi $(n-1)$ jer je vodeći koeficijent polinoma $\frac{1}{2^{n-1}}T_n(x)$ i $P(x)$ jednak 1. Vrijedi i

$$\frac{1}{2^{n-1}}T(x_k) = (-1)^k, \qquad |P_n(x_k)| \leq 1.$$

Zbog toga se predznak od $Q(x_k)$ podudara s predznakom od $T_n(x_k)$ ili je $Q(x_k) = 0$. Kod prvog slučaja imamo da $Q(x)$ poprima vrijednosti suprotnih predznaka u krajnjih točkama svakog intervala $[x_{k+1}, x_k]$. Zato $Q(x)$ ima nultočku u svakom od tih intervala. Ako je $Q(x_k) = 0$, treba se napraviti malo detaljnija analiza. U tom je slučaju ili nultočka dvostruka, pa $Q(x_k)$ ima dvije nultočke na intervalu $[x_{k+1}x_k]$, ili u jednom od susjednih intervala taj polinom posjeduje još jednu nultočku. Razlog tomu je što u točkama x_{k+1} i x_{k-1} polinom $Q(x)$ ima vrijednosti istog predznaka. Zbog toga što ima n intervala koji se promatraju, možemo zaključiti da polinom $Q(x)$ ima barem n nultočaka na intervalu $[-1, 1]$. Njegov stupanj je $(n-1)$ i zato je on identični jednak nuli. To znači da vrijedi tvrdnja

$$P(x) = \frac{1}{2^{n-1}}T_n(x).$$

8.2. Čebiševljevi polinomi druge vrste

Čebiševljev polinom druge vrste n-tog stupnja U_n, rješenje je diferencijalne jednadžbe

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

Za taj polinom vrijedi

$$U_n(x) = \frac{\sin((n+1)\arccos(x))}{\sin(\arccos(x))},$$

gdje je $x \in [-1, 1]$ i $n \in \mathbb{N}_0$. Čebiševljevi polinomi druge vrste definirani su sa

$$\sin(n+1)\alpha + \sin(n-1)\alpha = 2\sin n\alpha\cos\alpha.$$

Ako uvedemo supstituciju

$$\alpha = \arccos x$$

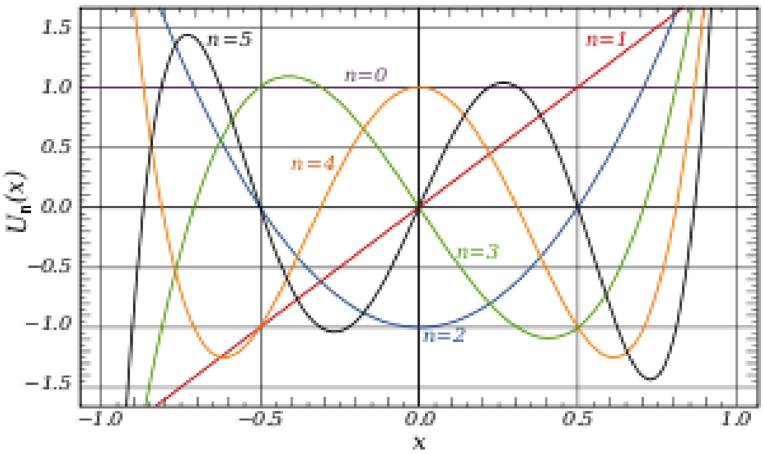
dobivamo rekurzivnu relaciju za Čebiševljeve polinome druge vrste koja glasi

$$U_{n+1}(x) - 2xU_n(x) + U_{n-1}(x) = 0,$$

uz uvjete $U_0(x) = 1$ i $U_1(x) = 2x$. Tako imamo nekoliko prvih Čebiševljevih polinoma druge vrste [26]

$$U_0(x) = 1$$
$$U_1(x) = 2x$$
$$U_2(x) = 4x^2 - 1$$
$$U_3(x) = 8x^3 - 4x$$
$$U_4(x) = 16x^4 - 12x^2 + 1$$
$$\vdots$$
$$U_8(x) = 256x^8 - 448x^6 + 260x^4 - 40x^2 + 1$$
$$U_9(x) = 512x^9 - 1024x^7 + 672x^5 - 160x^3 + 10x.$$

Sljedeća slika prikazuje graf nekoliko prvih Čebiševljevih polinoma druge vrste.



Slika 10: Čebiševljevi polinomi druge vrste
Izvor: [*https://en.wikipedia.org/wiki/Chebyshev_polynomials*]

Čebiševljeve polinome prve i druge vrste možemo povezati sljedećim relacijama:

$$T_n(x) = U_n(x) - xU_{n-1}(x),$$

$$T_n(x) = xU_{n-1}(x) - U_{n-2}(x),$$

$$T_n(x) = \frac{1}{2}\big(U_n(x) - U_{n-2}(x)\big),$$

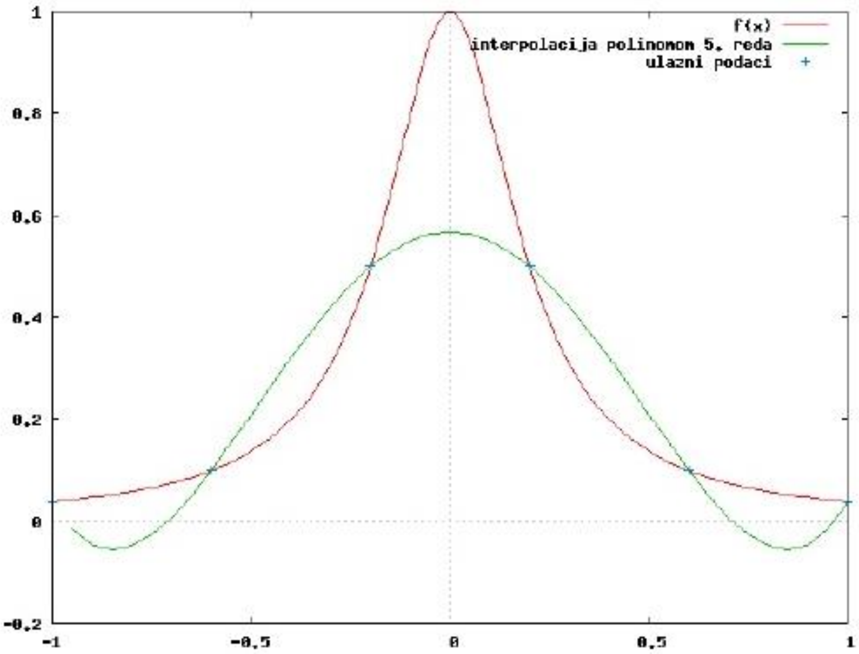
$$U_{n-1}(x) = \frac{1}{1-x^2}[xT_n(x) - T_{n+1}(x)].$$

9. Spline interpolacija

Interpolacija polinomima nije uvijek najbolje rješenje, pogotovo ako su u pitanju polinomi visokog stupnja, koji se u praksi ni ne bi smjeli koristiti. Na sljedećoj slici možemo vidjeti loša svojstva interpolacije funkcije polinomom. Funkcija koja se interpolira je

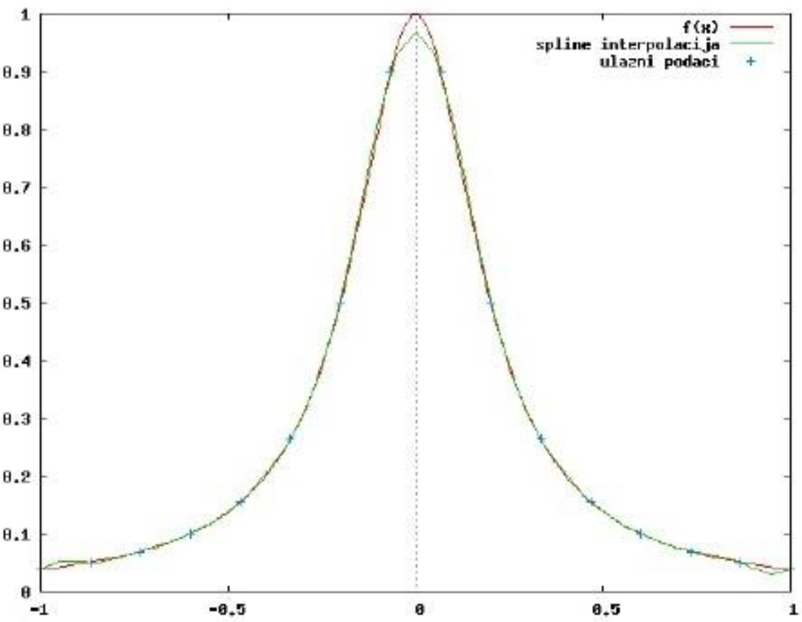
$$f(x) = \frac{1}{1 + 25x^2},$$

na intervalu $[-1, 1]$, a polinom je $5.$ stupnja. [31]



Slika 11: Interpolacija funkcije polinomom
Izvor: $[31]$

Vidimo da rezultat nije baš zadovoljavajući. Za neke točke je greška koju radi interpolacijski polinom veća od samog rezultata funkcije. Ako povećavamo stupanj interpolacijskog polinoma, nećemo dobiti bolje rezultate. U takvim situacijama najbolje je rješenje spline interpolacija koja interpolira podatke lokalno s polinomima, ali i globalno ostaje blizu podacima, a tako i pravoj vrijednosti funkcije. To radi tako da povezuje interpolirane segmente na način da su funkcija, kao i njezine prva i druga derivacija neprekidne. U odnosu na prethodnu sliku, povećali smo broj ulaznih podataka kod spline aproksimacije i rezultat je puno bolji, a nema ni oscilacija kao kod polinomne interpolacije.



Slika 12: Interpolacija funkcije spline-om
Izvor: [31]

Spline interpolacija je po dijelovima polinomna interpolacija funkcije f gdje na svakom njezinom podintervalu vrijedi

$$\varphi|_{[x_{k-1},x_k]} = p_k, \qquad k = 1, 2, \dots, n,$$

i gdje su p_k polinomi niskog, ali fiksnog stupnja. Kod spline-a vrijedi i

$$a = x_0 < x_1 < \cdots < x_n = b$$

gdje je $[a, b]$ interval koji je podijeljen na podintervale. Pretpostavimo da se na svakom podintervalu $[x_{k-1}, x_k]$ koristi polinom m-tog stupnja. Svaki takav polinom p_k određen je s $(m+1)$-im koeficijentom. To znači da ukupno moramo odrediti koeficijente polinoma p_k u n podintervala, to jest

$$(m+1) \cdot n$$

koeficijenata. Interpolacijski uvjeti su

$$\varphi(x_k) = f_k, \qquad k = 0, \dots, n.$$

To znači da za svaki polinom imamo po 2 uvjeta

$$p_k(x_{k-1}) = f_{k-1}$$

$$p_k(x_k) = f_k,$$

za $k = 1, \dots, n$. Sveukupno je to $2n$ interpolacijskih uvjeta. Postavljanjem tih interpolacijskih uvjeta osigurali smo neprekidnost funkcije φ, jer je [22]

$$p_{k-1}(x_{k-1}) = p_k(x_{k-1}), \qquad k = 2, \dots, n.$$

Pošto je interpolacijskih uvjeta $2n$ i ako moramo pronaći $(m+1)\cdot n$ koeficijenata, bez dodatnih uvjeta to možemo napraviti samo za $m=1$, odnosno za po dijelovima linearnu interpolaciju. Za $m>1$ moraju se dodati još neki uvjeti na glatkoću interpolacijske funkcije φ u čvorovima interpolacije. [22]

9.1. Linearni spline

Glavna ideja korištenja linearnog spline-a, odnosno po dijelovima linearne interpolacije leži u tome da umjesto jednog polinoma visokog stupnja koristimo više polinoma nižeg stupnja. Pod polinomima nižeg stupnja najčešće podrazumijevamo polinome prvog stupnja. Korištenjem takvih polinoma dobivamo linearnu interpolaciju koja je globalno neprekinuta i ne diferencijabilna u poznatim točkama $x_0, \dots, x_n$ što joj znatno ograničava upotrebu. Ova metoda nije komplicirana, naprotiv dosta je jednostavna za korištenje, no njezin nedostatak je u tome što je relativno neprecizna. [22]

Polinom p_k stupnja 1 je na svakom podintervalu $[x_{k-1}, x_k]$ jedinstveno određen. Takav polinom zapisujemo uzimajući u obzir početnu točku tog intervala u obliku:

$$p_k(x) = c_{0,k} + c_{1,k}(x - x_{k-1}),$$

gdje je $x \in [x_{k-1}, x_k]$, za $k = 1,2, \dots, n.$ Isto tako prethodni polinom možemo zapisati i u Newtonovoj formi gdje će izgledati:

$$p_k(x) = f[x_{k-1}] + f[x_{k-1}, x_k] \cdot (x - x_{k-1}).$$

Iz usporedbe prethodna dva polinoma zapisana u različitom obliku možemo zaključiti i vidjeti sljedeće:

$$c_{0,k} = f[x_{k-1}] = f_{k-1}, \qquad c_{1,k} = f[x_{k-1}, x_k] = \frac{f_k - f_{k-1}}{x_k - x_{k-1}}, \qquad k = 1,2, \dots, n.$$

Dakle, kako bismo aproksimirali neku vrijednost funkcije f u jednoj točki za koju vrijedi $x \in [a, b]$ trebamo prvo pronaći između kojih čvorova se točka x nalazi, to jest za koji će k vrijediti $x_{k-1} \leq x \leq x_k$. Nakon što smo pronašli k, možemo izračunati koeficijente pripadnog linearnog polinoma. [22]

9.1.1. Algoritam za linearni spline

Kako bismo pronašli prethodno spomenuti podinterval $x \in [x_{k-1}, x_k]$, koristimo algoritam binarnog pretraživanja. Binarno pretraživanje funkcionira tako da prvo pronađe centralni element, zatim odbaci jednu polovicu sortirane liste, pa pretražuje drugu polovicu. To pretraživanje se nastavlja sve dok se ne pronađe traženi element ili ne ponestane elemenata u listi. Trajanje tog algoritma je proporcionalno s $\log_2(n)$, gdje je n broj elemenata. Srž tog algoritma je:

$dno = 0;$

$vrh = n;$

$dok\ je\ (vrh - dno) > 1\ radi\ \{$

$\quad sredina = (dno + vrh)/2;$

$\quad ako\ je\ x < polje[sredina]\ onda$

$\quad\quad vrh = sredina;$

$\quad inače$

$\quad\quad dno = sredina;$

$\};$

Nakon provedenog binarnog pretraživanja dobivamo $x_{k-1} = polje[dno], x_k = polje[vrh]$ gdje je $polje$ polje svih čvorova koje promatramo, a pretpostavljeno je da su svi čvorovi poznati. Broj čvorova je n.

$c_0 = funkcija(x_{k-1});$

$c_1 = \left(funkcija(x_k) - funkcija(x_{k-1})\right)/\ (x_k - x_{k-1});$

$ispiši(c_0\ "+"\ c_1\ "(x - "x_{k-1}")"\ ;$

Naposljetku je ispisan interpolacijski polinom.

9.1.2. Ocjena greške linearnog spline-a

Ako imamo funkciju f koja je klase $C^2[a,b]$, a $[a,b]$ je interval na kojem aproksimiramo, onda možemo dobiti grešku te interpolacije. Ona je maksimalna pogreška od n linearnih interpolacija. Ocjena greške linearne interpolacije na podintervalu $[x_{k-1}, x_k]$ iznosi

$$|f(x) - p_k(x)| \leq \frac{M_2^k}{2!} |\omega(x)|,$$

gdje je

$$\omega(x) = (x - x_{k-1})(x - x_k), \qquad M_2^k = \max_{x \in [x_{k-1}, x_k]} |f''(x)|.$$

Ako je razmak između susjednih čvorova $h_k = x_k - x_{k-1}$, može se definirati maksimalni razmak susjednih čvorova

$$h = \max_{1 \leq k \leq n} \{h_k\}$$

pa je ocjena greške linearne interpolacije na čitavom intervalu $[a,b]$

$$|f(x) - \varphi(x)| \leq \frac{M_2}{2!} \frac{h^2}{4} = \frac{1}{8} M_2 \cdot h^2.$$

To znači da, ako ravnomjerno povećavamo broj čvorova, tako da $h \to 0$, onda će i maksimalna greška težiti u 0. [22]

Ovdje ćemo vidjeti primjer linearnog spline-a za funkciju $f(x) = \sin x$ na segmentu $[0, 2\pi]$. Neka je $f(x) = \sin x$ za $[0, 2\pi]$. Segment će se dijeliti redom na $n = 1, 2, \dots, 40$ podintervala.

Primjer za $n = 15$ u točki $x = 1$.

$$h = \frac{2\pi - 0}{15} = \frac{2\pi}{15}$$

$$x_* \in [x_{k-1}, x_k], \qquad x_* = 1$$

$\Longrightarrow 0 + (k-1)h \ \leq 1 \leq 0 + k \cdot h \ /\div h$

$(k-1) \leq 2.388 \leq k$

$\Longrightarrow k - 1 = 2,\ k = 3.$

Tablica 16: Izračun koeficijenata linearnog spline-a za $f(x) = sin\, x$

	$f[x_{k-1}]$	$f[x_{k-1}, x_k]$
$x_{k-1} \approx 0.837758041$	0.743144825	0.4963526
$x_k \approx 1.256637061$	0.951056516	

$\Longrightarrow p_3(x) = 0.743144825 + 0.4963526(x - 0.837758041)$

$p_3(1) \approx 0.823674043$

$\sin(1) \approx 0.841470984$

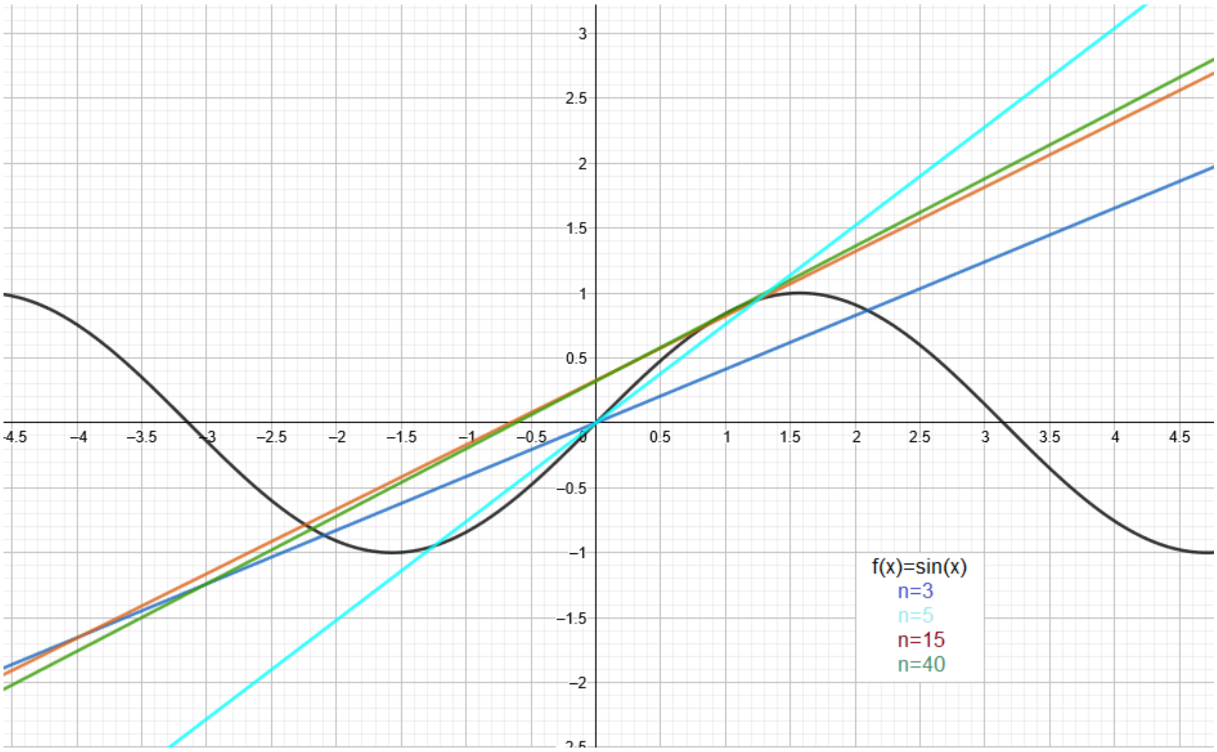
Greška je $|f(x) - p_3(x)| \approx 0.841470984 - 0.823674043 \approx 0.017796941.$

Analogno dalje dobivaju se sljedeći rezultati za $x = 1$:

Tablica 17: Usporedba linearnih spline-ova za $f(x) = sin\ x$

n	$\sin(x)$	SPLAJN	$p_k(x)$	$\lvert \sin(x) - p_k(x)\rvert$
3	0.841470984	0 + 0.413497 (x - 0)	0.413497	0.427974
4		0 + 0.63662 (x - 0)	0.63662	0.204851
5		0 + 0.756827 (x - 0)	0.756827	0.0846443
…		…	…	…
14		0.781832 + 0.430251 (x - 0.897598)	0.82589	0.0155809
15		0.743144825+0.4963526 (x – 0.837758041)	0.82367404	0.017796941
16		0.707107 + 0.552007 (x - 0.785398)	0.825569	0.0159024
…		…	…	…
27		0.802123 + 0.498873 (x - 0.930842)	0.836624	0.0048469
28		0.781832 + 0.530916 (x - 0.897598)	0.836198	0.00527253
29		0.762162 + 0.56009 (x - 0.866646)	0.836852	0.00461879
…		…	…	…
37		0.750672 + 0.593919 (x - 0.849079)	0.840307	0.00116393
38		0.837166 + 0.475405 (x - 0.992082)	0.840931	0.000540164
39		0.822984 + 0.499459 (x - 0.966644)	0.839644	0.00182715
40		0.809017 + 0.521962 (x - 0.942478)	0.839041	0.00242963

Dalje slijedi prikaz linearnih splajnova za $f(x)$. Prikazana je sama funkcija $f(x) = \sin(x)$, kao i rezultati aproksimacija te funkcije linearnim spline-om u stupnjevima $3, 5, 15$ i 40.



Slika 13: Prikaz linearnih spline-ova za funkciju $f(x) = sin(x)$

Iz priložene tablice i grafa možemo zaključiti da ravnomjernim povećavanjem broja čvorova tako da $h \to 0$, dobivamo također da maksimalna greška teži u 0.

9.1.3. Nedostaci linearnih spline-ova

Nakon što smo upoznali i vidjeli na primjeru samu primjenu linearnih spline-ova, možemo donijeti nekoliko zaključaka oko nedostataka po dijelovima linearne interpolacije. Prvo što možemo uočiti jest da nam je potrebno dosta podintervala kako bi dobili relativno umjerenu točnost aproksimacije. Isto tako već smo prije spomenuli, a možemo i vidjeti da sama funkcija nije glatka kao što je to kod recimo po dijelovima kubične interpolacije. Prema grafovima u primjeru može se vidjeti da je funkcija neprekidna te kao takva daje kao rezultat relativnu nepreciznost. Možemo uz to spomenuti i ocjenu greške na primjeru. Uzmimo da je maksimalni razmak po svim podintervalima 0.01 za nekih 100 linearnih interpolacija. Tada će greška aproksimacije biti reda 10^{-4} što potvrđuje ranije spomenutu nepreciznost. Zbog spomenutih razloga, odnosno nedostataka na svakom podintervalu koristimo polinome viših stupnjeva.

10. B-spline

Kada govorimo o B-spline krivuljama govorimo o grupi neprekinutih, parametarski zadanih polinoma po dijelovima povezanih točkama koje zovemo kontrolne točke. B-spline k-tog reda je po dijelovima polinom $(k-1)$-og stupnja, s $(k-2)$ kontinuiranih, odnosno neprekinutih derivacija u čvorovima. Kad nam je zadan skup od $(n+1)$-e kontrolne točke d_i, $i = 0, 1, \ldots, n$, gdje je $d_i \in \mathbb{R}^2$, B-spline krivulju reda k možemo definirati kao

$$P(x) = \left(N_{0,k}(x) \cdot d_0\right) + \left(N_{1,k}(x) \cdot d_1\right) + \cdots + (N_{n,k}(x) \cdot d_n) = \sum_{i=0}^{n} N_{i,k}(x) \cdot d_i.$$

$N_{i,k}$ je osnovna , odnosno bazična B-spline funkcija k-tog stupnja, $k = 1, 2, \ldots., n+1$. Imamo $(n+1)$-u takvu funkciju. B-spline baznu funkciju $N_{i,k}$ reda k definiramo na način:

$$k = 1: \quad N_{i,1}(x) = \begin{cases} 1, & za\ x_i \leq x < x_{i+k} \\ 0, & \text{inače} \end{cases},$$

$$k > 1: \quad N_{i,k}(x) = \frac{x - x_i}{x_{i+k-1} - x_i} N_{i,k-1}(x) + \frac{x_{i+k} - x}{x_{i+k} - x_{i+1}} N_{i+k,k-1}(x).$$

Velika prednost za interpolaciju i modifikaciju krivulje je to što su sve funkcije definirane samo lokalno. To znači da koeficijenti polinoma, a samim time i cijela krivulja zavise samo od nekoliko kontrolnih točaka. Tako imamo, ako je $k = 4$, da je prvi segment krivulje definiran pomoću prve četiri kontrolne točke, a zadnji segment krivulje je definiran pomoću posljednje četiri kontrolne točke. [32]

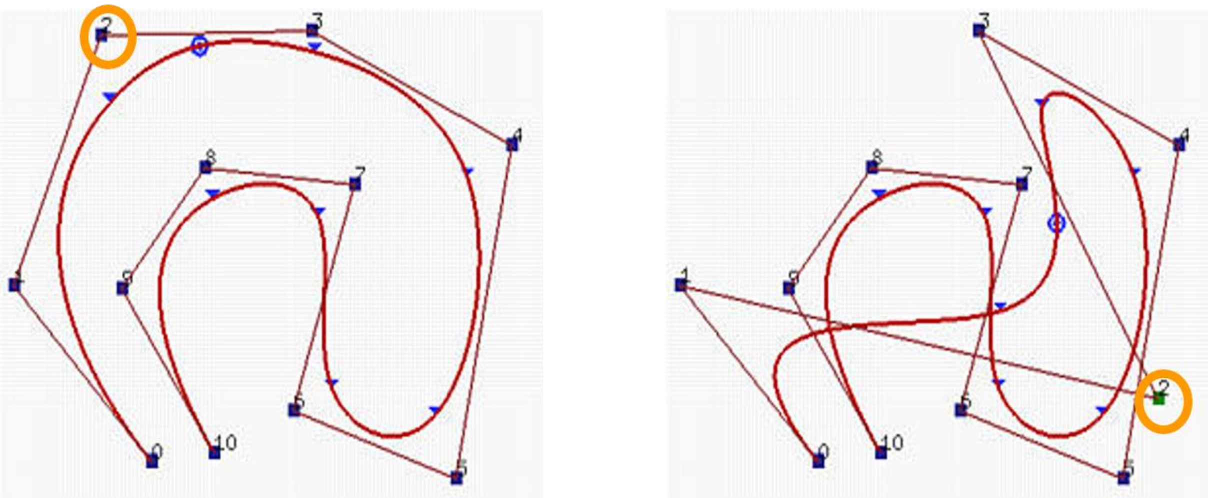
B-spline krivulja sastoji se od onoliko polinomskih dijelova, odnosno segmenata krivulje koliko je kontrolnih točaka minus stupanj polinoma. Za $(n+1)$-u kontrolnu točku i k kao stupanj polinoma bit će $n-k+1$ dijelova krivulje. Za čvorove te krivulje x_j, $j = i, \ldots i+k$ vrijedi $x_i < x_{i+1} < \cdots < x_{i+k}$. Takvih čvorova postoji $m+1$, od 0 do m. [33]

Kod B-spline krivulja postoji još jedan termin koji je važan kod njihovog izračuna, a on je vektor čvorova kojeg označavamo sa U. Sastoji se od svih vrijednosti čvorova $U = \{x_j\}$. Vrijednosti čvorova se ne moraju striktno povećavati tako da mogu postojati i višestruki čvorovi. Lomovi ili skokovi u krivulji postižu se korištenjem višestrukih čvorova. Sljedeća relacija pokazuje vezu između vektora čvorova, broja točaka i stupnja krivulje

$$U = \{\underbrace{0, \ldots 0}_{k+1}, \underbrace{1, 2, 3, (m-2k-1)}_{m-2k-1}, \underbrace{(m-2k) \ldots (m-2k)}_{k+1}\}$$

gdje je $n+1$ broj kontrolnih točaka, k stupanj krivulje, a $m+1$ broj vrijednosti u vektoru čvorova koje možemo izračunati kao $m = n+k+1$. Broj segmenata, to jest dijelova krivulje možemo izračunati, osim na spomenuti način $n-k+1$, i preko $m-2k$. [34]

Kao što smo već rekli, glavna prednost B-spline-ova je lokalna kontrola oblika krivulje. To ujedno znači i da se mogu dodavati kontrolne točke bez da se ne poveća stupanj krivulje. Isto tako, izbjegnut je i problem spajanja dijelova krivulje jer su dozvoljene samo one krivulje koje imaju zahtijevanu kontinuiranost na čvorovima. Rekli smo da je svaki segment B-spline krivulje pod utjecajem k kontrolnih točaka, npr. ako je $k = 3$, $P(x) = N_{i-1,k}p_{i-1} + N_{i,k}p_i + N_{i+1,k}p_{i+1}$, pa se tako pomicanjem neke kontrolne točke utječe na izgled krivulje samo u neposrednoj blizini te točke, to jest na intervalu $[x_i, x_{i+1})$. Zbog toga je i vrijeme računanja koeficijenata smanjeno. Na sljedećoj slici ćemo vidjeti to svojstvo. Pomaknuli smo točku P_2 i krivulja se promijenila samo u blizini te točke, to jest samo prvi, drugi i treći segment te krivulje mijenjaju svoj oblik, dok ostali segmenti krivulje ostaju u svom prvobitnom položaju. [35]



Slika 14: Pomicanje točke kod B-spline-a
Izvor: [*36*]

Bazične B-spline funkcije mogu biti neracionalne ili racionalne, neujednačene i ujednačene. Kod ujednačenih, odnosno uniformnih B-spline-ova razmak između dva susjedna čvora je konstantan, to jest $x_{i+1} - x_i = const$. Definiranje B-spline-ova i bazičnih funkcija već smo objasnili pa ovdje ne trebamo jer se to definiranje odnosi na ujednačene B-spline-ove jer se smatra, ako se govori samo o B-spline-ovima, da su oni ujednačeni. [35]

Kod uniformnih B- spline-ova postoje dvije različite vrste čvorova, to jest vektora čvorova, a to su neperiodički i periodički. Kod neperiodičkih čvorova prvi i zadnji čvor su duplicirani k puta, npr. $(0, 0, 1, 2, 3, 4, 5, 5)$ i krivulja prolazi kroz prvu i zadnju kontrolnu točku. Kod te vrste čvorova stupanj spline-a je k, broj kontrolnih točaka je $n + 1$, broj čvorova je $m = n + k + 1$. [36]

Tako možemo navesti primjer: Za stupanj $k = 2$ i broj kontrolnih točaka $= 4$ imamo $k = 2, n = 3$ (jer je $n + 1 = 4$), a broj čvorova je $m + 1 = 7$ ($m = n + k + 1 = 6$). Za taj primjer vektor čvorova iznosi $(0, 0, 1, 2, 3, 4, 4)$. Kod periodičkih čvorova prvi i zadnji čvor nisu duplicirani, npr. $(0, 1, 2, 3)$ i krivulja ne prolazi zadnjom točkom. Kod uniformnih spline-ova se obično uzima da je $x_0 = 0$, a $x_m = 1$ tako da je domena krivulje interval $[0, 1]$. [36]

Neujednačeni, odnosno neuniformni B-spline-ovi ne moraju imati konstantan razmak između čvorova. Ima više prednosti ove vrste spline-ova u odnosu na uniformne. Kontinuiranost u nekoj točki može biti smanjena iz druge derivacije na prvu ili na nultu. Ako je kontinuiranost smanjena na nultu, onda krivulja interpolira tu kontrolnu točku, ali bez nekih neželjenih efekata koji bi se javili kod uniformnih B-spline-ova. Također, prva i zadnja kontrolna točka mogu biti točno interpolirane, bez da se istovremeno uvede ravna linija na segmentu. Kod neujednačenih B-spline-ova možemo dodati neku kontrolnu točku ili novi čvor pa se s tim krivulja preoblikuje, a to nije bilo moguće kod ujednačenih B-spline-ova. [35]

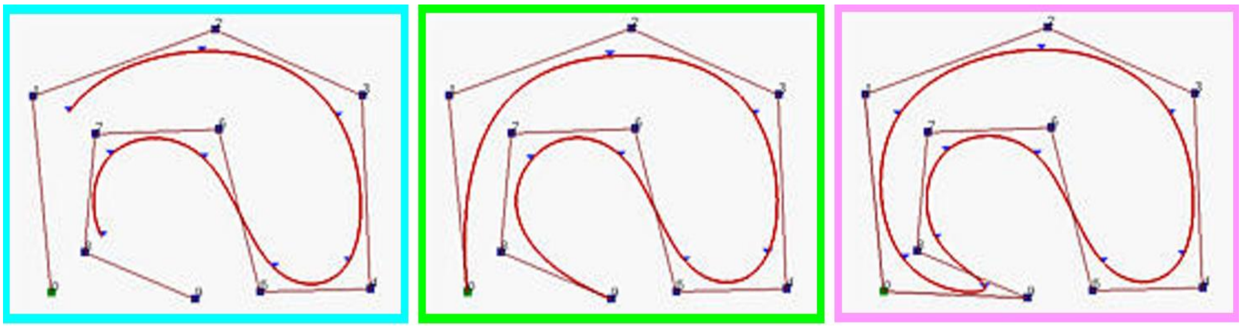
Postoje i neujednačeni racionalni B-spline-ovi koje zovemo NURBS. Kod te vrste krivulja za opisivanje jedne kontrolne točke ne koristimo 3 parametra (x, y, z), nego 4 (x, y, z, w). To radimo zbog mogućnosti točnog prikazivanja nekih krivulja, kao što su krug, elipsa, ali i zbog povećane kontrole nad oblikom drugih krivulja. Dodana koordinata w zove se težina kontrolne točke. Svaka kontrolna točka obično ima težinu 1, što podrazumijeva da one imaju jednak utjecaj na oblik same krivulje. Međutim, ako povećamo težinu određene kontrolne točke njoj se automatski poveća utjecaj i to ima za efekt „povlačenje“ krivulje prema toj kontrolnoj točki. Krivulje definirane na ovaj način, gdje svaka kontrolna točka ima svoju težinu, nazivamo racionalnim krivuljama. NURBS krivulju reda k možemo definirati kao

$$Q(x) = \frac{\sum_{i=0}^{m} W_i P_i N_{i,k}(x)}{\sum_{i=0}^{m} W_i N_{i,k}(x)}$$

gdje je $x \in [x_{k-1}, x_{m+k}]$, $(x_0, \dots, x_{m+k})$ skup čvorova, $(P_0, \dots, P_m)$ skup kontrolnih točaka, a $(w_0, \dots, w_m)$ skup težina kontrolnih točaka. [35]

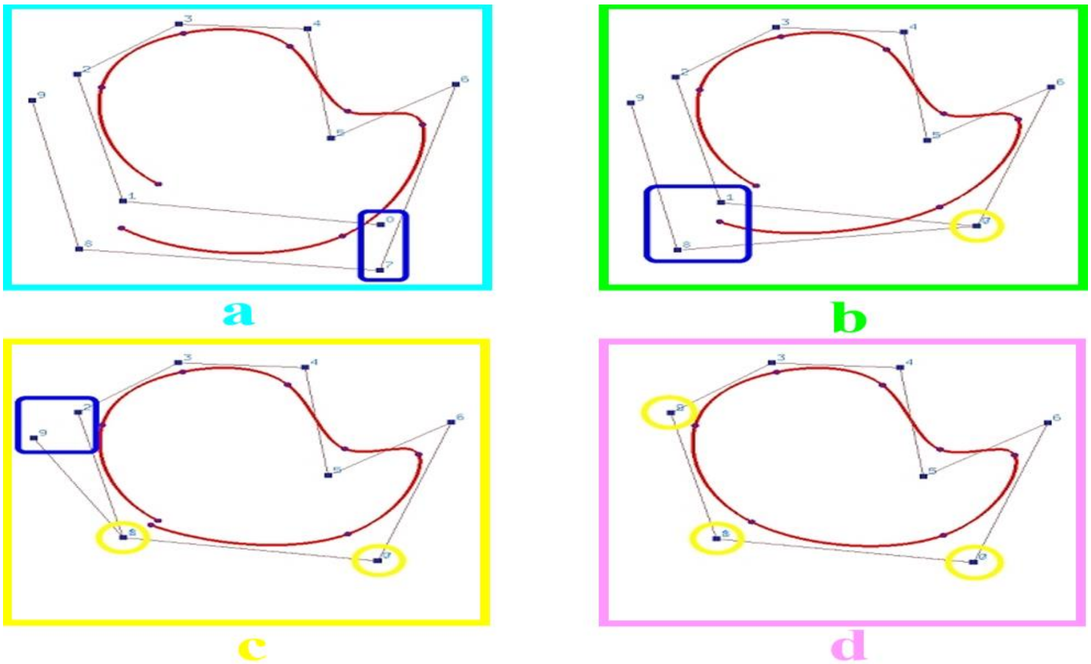
Što se tiče same strukture B-spline-ova, one mogu biti podijeljene na tri vrste: otvorene B-spline krivulje, stegnute (engl. *clamped*) B-spline krivulje i zatvorene B-spline krivulje. [37]

Otvorene B-spline krivulje karakterizira to da krivulja ne dodiruje prvu i zadnju kontrolnu točku. Za takvu vrstu krivulja domena je $[x_k, x_{m-k}]$. Ako je npr. B-spline krivulja stupnja 6 $(k = 6)$ i definirana je s 14 kontrolnih točaka $(n = 13)$, tada je broj čvorova $m + 1 = 21$ $(m = n + k + 1 = 20)$. Ako je vektor čvorova uniformni, to jest ujednačen, on iznosi $U = \{0, 0.05, 0.1, 0.15, \dots, 0.9, 0.95, 1\}$. Ta otvorena krivulja definirana je na $[x_k, x_{m-k}] = [x_6, x_{14}] = [0.3, 0.7]$. Kod stegnutih B-spline-ova krivulja je tangencijalna na prvoj i zadnjoj točki polinomske linije. Kod zatvorenih B-spline-ova, ako ponavljamo neke čvorove i kontrolne točke, krivulja će postati zatvorena. Tada se početak i kraj krivulje spajaju formirajući zatvorenu petlju. Sljedeća slika prikazuje te tri vrste B-spline-ova. Najprije je otvorena, pa stegnuta i na kraju zatvorena B-spline krivulja. [37]



Slika 15: Vrste B-spline-ova prema strukturi
Izvor: [*37*]

Zatvorenu B-spline krivulju možemo konstruirati i tako da postojećoj krivulji, koja je otvorena, promijenimo oblik, to jest spojimo neke kontrolne točke. Tako sljedeća slika prikazuje konstrukciju krivulje iz otvorene u zatvorenu tako da spojimo kontrolne točke 0 i 7, 1 i 8, te 2 i 9.



Slika 16: Konstruiranje B-spline-a iz otvorenog u zatvoreni
Izvor: [*37*]

11. Implementacija algoritama metoda aproksimacije/interpolacije

11.1. Implementacija algoritma linearnog spline-a

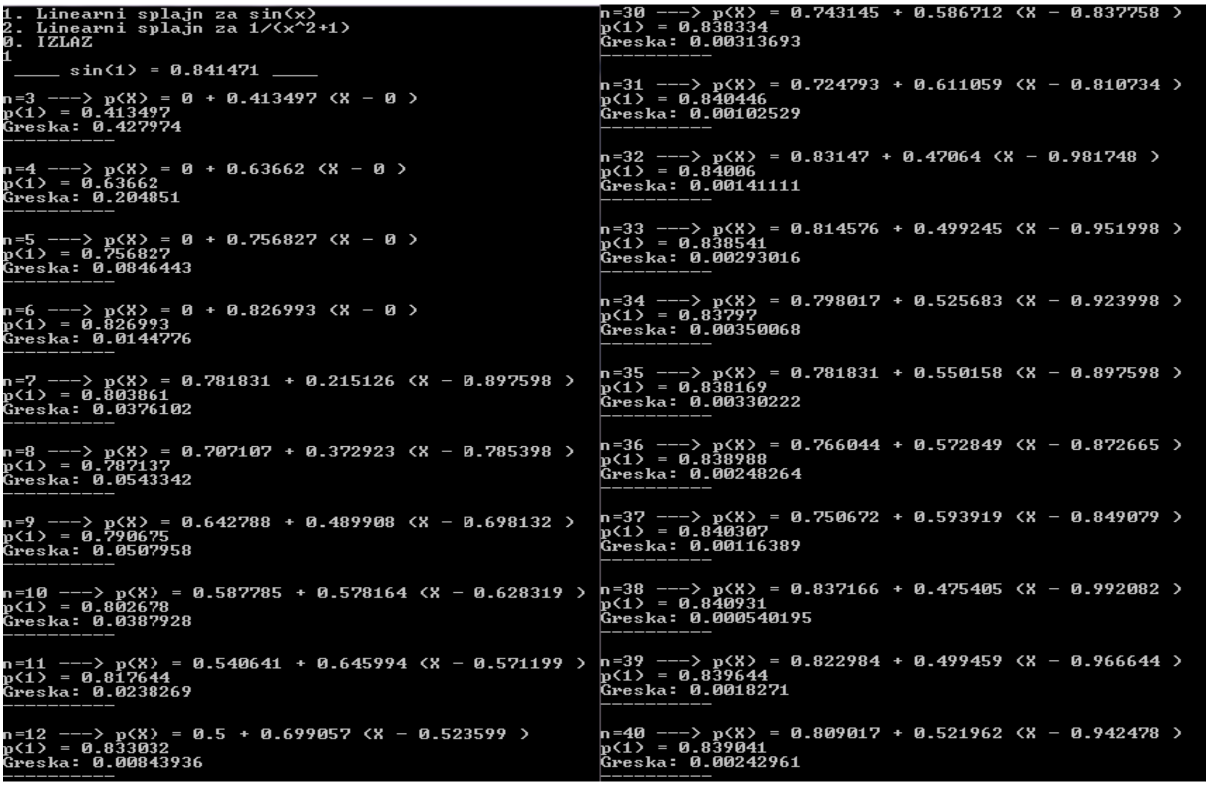
Na sljedećem primjeru vidjet ćemo implementaciju algoritma linearnog spline-a za dvije funkcije, a to su $\sin x$ i $1/(x^2+1)$. Algoritam je napisan u programskom jeziku C++.

```
#include < iostream >
#include < cmath >
using namespace std;
double podinterval[2];

void bin_pretrazivanje(unsigned int n, double a, double b, double x){
        double p[n + 1];
        unsigned int vrh = n, dno = 0, sredina;
        double h = (b − a)/n;
        for(int i = 0; i <= n; i + +) p[i] = a + i · h;
        while((vrh − dno > 1)){
                sredina = (vrh + dno)/2;
                if(x < p[sredina]) vrh = sredina;
                else dno = sredina;
        }
        double polje[2];
        podinterval[0] = p[dno];
        podinterval[1] = p[vrh];
}
double f(double x){
        return sin(x);
}
double g(double x){
        return 1/(x · x + 1);
}
```

```
int main(){

int izbor;

do{

cout << "1.Linearni splajn za sin(x)" << endl << "2.Linearni splajn za 1/ (x^2 + 1)" << endl << "0.IZLAZ" << endl;

cin >> izbor;

switch(izbor){

	case 1:

		cout << "____sin(1) = " << f(1) << "____" << endl;

		for(int i = 3; i <= 40; i++){

			bin_pretrazivanje(i, 0, 6.283185307, 1);

			cout << endl;

			double pk = f(podinterval[0]) + ((f(podinterval[1]) - f(podinterval[0]))/(podinterval[1] - podinterval[0])) * (1 - podinterval[0]);

			cout << "n = " << i << " ---> p(X) = " << f(podinterval[0]) << " + " << (f(podinterval[1]) - f(podinterval[0]))/(podinterval[1] - podinterval[0]) << " (X - " << podinterval[0] << " )" << endl;

			cout << "p(1) = " << pk << endl;

			cout << "Greska: " << abs(f(1) - pk) << endl << "-----------" << endl;

		}

		break;

	case 2:

		cout << "____g(1.2) = " << g(1.2) << "____" << endl;

		for(int i = 1; i <= 40; i++){

			bin_pretrazivanje(i, -5, 5, 1.2);

			cout << endl;

			double pk = g(podinterval[0]) + ((g(podinterval[1]) - g(podinterval[0]))/(podinterval[1] - podinterval[0])) * (1.2 - podinterval[0]);

			cout << "n = " << i << " ---> p(X) = " << g(podinterval[0]) << " + " << (g(podinterval[1]) - g(podinterval[0]))/(podinterval[1] - podinterval[0]) << " (X - " << podinterval[0] << " )" << endl;

			cout << "p(1.2) = " << pk << endl;
```

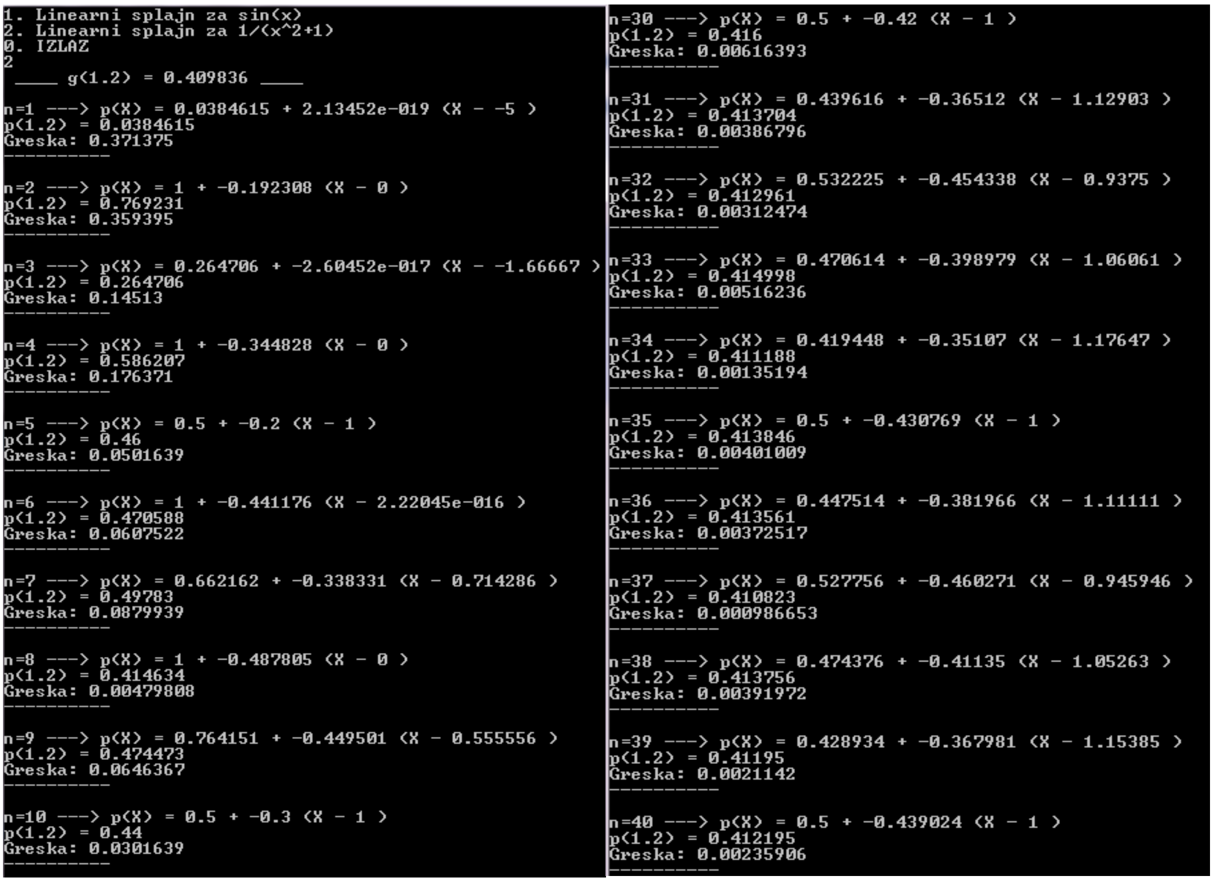
```
cout << "Greska: " << abs(g(1.2) - pk) << endl << "-----
-----" << endl;
        }
        break;
}
}while(izbor != 0);
cout << "Kraj programa! " << endl;
system ("pause");
}
```

Ako napišemo broj koji nije $0, 1$ ili 2, ponovno će se pojaviti početni izbornik. U samom algoritmu imamo funkciju binarnog pretraživanja koje smo već prije objasnili kako funkcionira, kao i funkcije koje vraćaju rezultat $sin\,x$ i $1/(x^2+1)$ za pojedini x. Te tri funkcije nam pomažu kod glavnog izračuna. Kod glavnog izračuna imamo odabir kojim biramo da li hoćemo izračun prve ili druge spomenute funkcije. Za funkciju $sin\,x$ ispisuju se rezultati za $n = 1, 3, \ldots, 40$, kao i vrijednost te funkcije za pojedini n u točki 1. Ispisuje se i greška pojedine aproksimacije. Isti slučaj je i s drugom funkcijom, samo što se za vrijednost funkcije uzima točka 1.2. Rezultate algoritma prikazat ćemo na sljedećim slikama.



```
1. Linearni splajn za sin(x)
2. Linearni splajn za 1/(x^2+1)
0. IZLAZ
1
 _____ sin(1) = 0.841471 _____

n=3 ---> p(X) = 0 + 0.413497 (X - 0 )
p(1) = 0.413497
Greska: 0.427974
----------

n=4 ---> p(X) = 0 + 0.63662 (X - 0 )
p(1) = 0.63662
Greska: 0.204851
----------

n=5 ---> p(X) = 0 + 0.756827 (X - 0 )
p(1) = 0.756827
Greska: 0.0846443
----------

n=6 ---> p(X) = 0 + 0.826993 (X - 0 )
p(1) = 0.826993
Greska: 0.0144776
----------

n=7 ---> p(X) = 0.781831 + 0.215126 (X - 0.897598 )
p(1) = 0.803861
Greska: 0.0376102
----------

n=8 ---> p(X) = 0.707107 + 0.372923 (X - 0.785398 )
p(1) = 0.787137
Greska: 0.0543342
----------

n=9 ---> p(X) = 0.642788 + 0.489908 (X - 0.698132 )
p(1) = 0.790675
Greska: 0.0507958
----------

n=10 ---> p(X) = 0.587785 + 0.578164 (X - 0.628319 )
p(1) = 0.802678
Greska: 0.0387928
----------

n=11 ---> p(X) = 0.540641 + 0.645994 (X - 0.571199 )
p(1) = 0.817644
Greska: 0.0238269
----------

n=12 ---> p(X) = 0.5 + 0.699057 (X - 0.523599 )
p(1) = 0.833032
Greska: 0.00843936
----------

n=30 ---> p(X) = 0.743145 + 0.586712 (X - 0.837758 )
p(1) = 0.838334
Greska: 0.00313693
----------

n=31 ---> p(X) = 0.724793 + 0.611059 (X - 0.810734 )
p(1) = 0.840446
Greska: 0.00102529
----------

n=32 ---> p(X) = 0.83147 + 0.47064 (X - 0.981748 )
p(1) = 0.84006
Greska: 0.00141111
----------

n=33 ---> p(X) = 0.814576 + 0.499245 (X - 0.951998 )
p(1) = 0.838541
Greska: 0.00293016
----------

n=34 ---> p(X) = 0.798017 + 0.525683 (X - 0.923998 )
p(1) = 0.83797
Greska: 0.00350068
----------

n=35 ---> p(X) = 0.781831 + 0.550158 (X - 0.897598 )
p(1) = 0.838169
Greska: 0.00330222
----------

n=36 ---> p(X) = 0.766044 + 0.572849 (X - 0.872665 )
p(1) = 0.838988
Greska: 0.00248264
----------

n=37 ---> p(X) = 0.750672 + 0.593919 (X - 0.849079 )
p(1) = 0.840307
Greska: 0.00116389
----------

n=38 ---> p(X) = 0.837166 + 0.475405 (X - 0.992082 )
p(1) = 0.840931
Greska: 0.000540195
----------

n=39 ---> p(X) = 0.822984 + 0.499459 (X - 0.966644 )
p(1) = 0.839644
Greska: 0.0018271
----------

n=40 ---> p(X) = 0.809017 + 0.521962 (X - 0.942478 )
p(1) = 0.839041
Greska: 0.00242961
----------
```

Slika 17: Rezultati algoritma linearnog spline-a za funkciju $sin\,x$



```
1. Linearni splajn za sin(x)
2. Linearni splajn za 1/(x^2+1)
0. IZLAZ
2
 ____ g(1.2) = 0.409836 ____

n=1 ---> p(X) = 0.0384615 + 2.13452e-019 (X - -5 )
p(1.2) = 0.0384615
Greska: 0.371375
----------

n=2 ---> p(X) = 1 + -0.192308 (X - 0 )
p(1.2) = 0.769231
Greska: 0.359395
----------

n=3 ---> p(X) = 0.264706 + -2.60452e-017 (X - -1.66667 )
p(1.2) = 0.264706
Greska: 0.14513
----------

n=4 ---> p(X) = 1 + -0.344828 (X - 0 )
p(1.2) = 0.586207
Greska: 0.176371
----------

n=5 ---> p(X) = 0.5 + -0.2 (X - 1 )
p(1.2) = 0.46
Greska: 0.0501639
----------

n=6 ---> p(X) = 1 + -0.441176 (X - 2.22045e-016 )
p(1.2) = 0.470588
Greska: 0.0607522
----------

n=7 ---> p(X) = 0.662162 + -0.338331 (X - 0.714286 )
p(1.2) = 0.49783
Greska: 0.0879939
----------

n=8 ---> p(X) = 1 + -0.487805 (X - 0 )
p(1.2) = 0.414634
Greska: 0.00479808
----------

n=9 ---> p(X) = 0.764151 + -0.449501 (X - 0.555556 )
p(1.2) = 0.474473
Greska: 0.0646367
----------

n=10 ---> p(X) = 0.5 + -0.3 (X - 1 )
p(1.2) = 0.44
Greska: 0.0301639
----------

n=30 ---> p(X) = 0.5 + -0.42 (X - 1 )
p(1.2) = 0.416
Greska: 0.00616393
----------

n=31 ---> p(X) = 0.439616 + -0.36512 (X - 1.12903 )
p(1.2) = 0.413704
Greska: 0.00386796
----------

n=32 ---> p(X) = 0.532225 + -0.454338 (X - 0.9375 )
p(1.2) = 0.412961
Greska: 0.00312474
----------

n=33 ---> p(X) = 0.470614 + -0.398979 (X - 1.06061 )
p(1.2) = 0.414998
Greska: 0.00516236
----------

n=34 ---> p(X) = 0.419448 + -0.35107 (X - 1.17647 )
p(1.2) = 0.411188
Greska: 0.00135194
----------

n=35 ---> p(X) = 0.5 + -0.430769 (X - 1 )
p(1.2) = 0.413846
Greska: 0.00401009
----------

n=36 ---> p(X) = 0.447514 + -0.381966 (X - 1.11111 )
p(1.2) = 0.413561
Greska: 0.00372517
----------

n=37 ---> p(X) = 0.527756 + -0.460271 (X - 0.945946 )
p(1.2) = 0.410823
Greska: 0.000986653
----------

n=38 ---> p(X) = 0.474376 + -0.41135 (X - 1.05263 )
p(1.2) = 0.413756
Greska: 0.00391972
----------

n=39 ---> p(X) = 0.428934 + -0.367981 (X - 1.15385 )
p(1.2) = 0.41195
Greska: 0.0021142
----------

n=40 ---> p(X) = 0.5 + -0.439024 (X - 1 )
p(1.2) = 0.412195
Greska: 0.00235906
----------
```

Slika 18: Rezultati algoritma linearnog spline-a za funkciju $1/(x^2+1)$

11.2. Implementacija algoritma Lagrangeovog oblika interpolacijskog polinoma

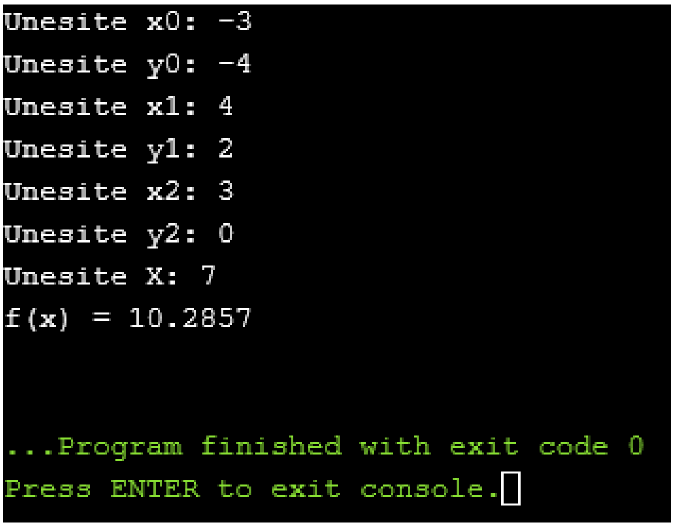
Ovdje ćemo vidjeti jednostavan primjer algoritma Lagrangeovog oblika interpolacijskog polinoma. U tom primjeru se unosi određeni broj točaka koje predstavljaju funkcije Lagrangeove baze i pomoću tih podataka algoritam izračunava $f(x)$ za neki x. Ovaj algoritam, kao i algoritam za linearni spline, napisan je u programskom jeziku C++. [38]

```
#include <cstdlib>
#include <iostream>
using namespace std;
int main()
{
  const int nmax = 3;
  float x[nmax] = {};
  float y[nmax] = {};
```

```
  for (int i = 0; i < nmax; i++)
  {
    cout << "Unesite x" << i << ": ";
    cin >> x[i];
    cout << "Unesite y" << i << ": ";
    cin >> y[i];
  }
  float X;
  float l[nmax] = {};
  float L = 0;
  cout << "Unesite X: ";
  cin >> X;

  for (int i = 0; i < nmax; i++)
  {
    l[i] = 1;
    for (int j = 0; j < nmax; j++)
    {
      if (i != j)
      {
        l[i] ·= (X - x[j])/(x[i] - x[j]);
      }
    }
  }
  for (int i = 0; i < nmax; i++)
  {
    L += y[i] · l[i];
  }
  cout << "f(x) = " << L << endl;

  // system("pause");
  return EXIT_SUCCESS;
}
```

U algoritmu imamo jednu varijablu pomoću koje definiramo koliko će biti funkcija Lagrangeovih baza, a to je varijabla $nmax$. Stavili smo da je ta varijabla 3 pa će toliko biti i funkcija

Lagrangeovih baza, to jest imati ćemo ℓ_0, ℓ_1 i ℓ_2. Te varijable spremaju se pomoću dva jednodimenzionalna polja. U jednom polju bit će uzlazno poredani x_i, a u drugom uzlazno poredani y_i. Nakon što smo unijeli sve x_i i y_i, algoritam može izračunati Lagrangeov oblik traženog interpolacijskog polinoma. Zatim moramo unijeti vrijednost X za koju će se računati vrijednost dobivene funkcije, odnosno $f(x)$. Sljedeća slika prikazuje dobiveni rezultat. Unijeli smo iste podatke koje smo imali kod jednog od zadatka za Lagrangeov oblik interpolacijskog polinoma pa se rezultati mogu lako provjeriti.



Slika 19: Rezultat Lagrangeovog algoritma

12. Zaključak

Metode aproksimacije sastavni su dio numeričke matematike. Pomoću aproksimacije skupa točaka mogu se odrediti nepoznati parametri funkcije koju aproksimiramo. Najčešće se za to koriste polinomi, ali mogu se koristiti i funkcije drugog oblika. Aproksimaciju možemo koristiti u mnogim slučajevima: za male grupe razbacanih podataka, za velike grupe podataka, kao i za lijepo grupirane podatke. Glavna ideja za određivanje aproksimativne, odnosno približne funkcije zasniva se na minimiziranju razmaka točaka podataka od aproksimativnog polinoma. Taj proces minimiziranja može se izvršiti na razne načine, ali za tu svrhu najčešće se koristi interpolacija i metoda najmanjih kvadrata. Aproksimacija je točnija što se uvrštavanjem određenih argumenata dobivaju vrijednosti bliske vrijednostima originalne funkcije. Razmatramo dva polazišta kod aproksimiranja: prvo da je funkcija $f(x)$ poznata, drugo da je nepoznata, to jest da su poznate samo njezine vrijednosti. Interpolacijski polinom neke funkcije može se koristiti, osim za procjenjivanje vrijednosti funkcije u točkama $x \neq x_i, x \in (x_0, x_n)$ i za zamjenu funkcije radi njenog približnog diferenciranja ili integriranja. Kod aproksimacije funkcija ne mora prolaziti točno kroz diskretne točke, a kod interpolacije funkcija prolazi točno kroz diskretne točke. Još jedna razlika aproksimacije i interpolacije je da u slučaju aproksimacije možemo postaviti vrijednost funkcije u bilo kojoj točki u svrhu dobivanja željene aproksimativne funkcije, a u slučaju interpolacije ulazne točke su određene i ne možemo ih mijenjati.

Ako znamo da je funkcija presložena za efikasnu procjenu, možemo izabrati nekoliko poznatih točaka podataka iz te složenije funkcije, izraditi tablicu i pokušati interpolirati te točke podataka kako bi konstruirali jednostavniju funkciju. Naravno, kada koristimo funkcije koje su jednostavnije od originalne, složenije funkcije za izračunavanje novih točaka podataka, vrlo vjerojatno nećemo dobiti isti rezultat kakav bi dobili da koristimo originalnu funkciju, već se javlja neka greška, ovisno o problemskoj domeni i korištenoj interpolacijskoj metodi. [1]

Na kraju možemo reći da je interpolacija korisna u slučajevima kada nam je poznat samo određen broj vrijednosti neke funkcije, a potrebne su nam vrijednosti te funkcije u drugim točkama, kao i u slučaju kad je originalna funkcija previše složena za izračunavanje. Vidjeli smo i da s povećanjem stupnja polinoma interpolacijski polinomi jako osciliraju, što je nepoželjno, pa zato interpolacijski interval obično dijelimo na podintervale i prelazimo na interpolaciju po dijelovima polinoma, to jest spline interpolaciju. Bez obzira na sve, ne može se izdvojiti općeniti postupak koji će se primjenjivati za sve slučajeve. Svaki slučaj moramo analizirati te odabrati najpovoljniju metodu koja će dati najbolje rezultate.

13. Dodatno

13.1. Metoda najmanjih kvadrata

U ovom smo radu više puta spominjali metodu najmanjih kvadrata, kao jednu od metoda koja se najviše koristi za aproksimiranje funkcija. Ovdje ćemo ju ukratko objasniti.

Ova metoda zasniva se na kriteriju u kojem se traži da suma kvadrata udaljenosti između izabranih točaka i pripadnih aproksimacija bude minimalna. Lagrangeovim oblikom interpolacijskog polinoma možemo dobro aproksimirati funkciju lokalno, u izabranim točkama. Međutim, izvan tih točaka aproksimacija može imati jako velike greške, pa se kao alternativa može koristiti metoda najmanjih kvadrata pomoću koje originalnu funkciju možemo aproksimirati nekom drugom funkcijom globalno tako da njihova udaljenost bude što manja, bez obzira na to što aproksimativna funkcija možda neće imati iste vrijednosti ni u jednoj točki, to jest bez obzira da li će se te dvije funkcije poklapati u točkama. [39]

Pretpostavimo da su vrijednosti zadane funkcije poznate samo u nekim točkama. Neka su $x_1, x_2, \ldots, x_n$ dane točke, a $f(x_1), f(x_2), \ldots, f(x_n)$ pripadne vrijednosti funkcije f. Trebamo pronaći onu funkciju g s neodređenim parametrima $\alpha, \beta, \ldots$ koja najbolje aproksimira originalnu funkciju f. To radimo na sljedeći način. Izračunamo sumu kvadrata razlika funkcija f i g

$$S = \sum_{k=1}^{n} [f(x_k) - g(x_k)]^2,$$

i onda tražimo parametre $\alpha, \beta \ldots$ iz uvjeta da S ima minimalnu vrijednost. Ako je originalna funkcija f zadana u svim točkama određenog segmenta $[a, b]$, onda se funkcija S definira pomoću integrala

$$S = \int_a^b [f(x) - g(x)]^2 dx.$$

Kao i kod drugih metoda, i za ovu metodu se za određivanje funkcije g mogu koristiti polinomi, eksponencijalne funkcije, itd. Na primjer ako želimo pronaći polinom prvog stupnja koji je najbliži funkciji f čije su nam vrijednosti poznate u točkama $x_1, x_2, \ldots, x_n$, onda je

$$S(\alpha, \beta) = \sum_{k=1}^{n} [f(x_k) - \alpha x_k - \beta]^2.$$

Nužan uvjet za ekstrem funkcije S je

$$\frac{\partial S}{\partial \alpha} = 0, \qquad \frac{\partial S}{\partial \beta} = 0,$$

dakle

$$\sum_{k=1}^{n}[f(x_k) - \alpha x_k - \beta]\, x_k = 0,$$

$$\sum_{k=1}^{n}[f(x_k) - \alpha x_k - \beta] = 0.$$

Kad se ovaj sustav jednadžbi sredi, dobije se

$$\alpha \sum_{k=1}^{n} x_k^2 + \beta \sum_{k=1}^{n} x_k = \sum_{k=1}^{n} f(x_k) x_k,$$

$$\alpha \sum_{k=1}^{n} x_k + n\beta = \sum_{k=1}^{n} f(x_k).$$

Iz njega se izračunaju parametri α i β. Iz prirode problema jasno je da funkcija S ima samo minimum, pa zato parametri α i β doista daju najbolju aproksimaciju funkcije f. Kada se funkcija g traži u eksponencijalnom ili logaritamskom obliku, onda se najčešće dobije sustav nelinearnih jednadžbi, koji je teže riješiti od linearnog. [39]

Primjer: Neka su zadane funkcija i točke: $f(x) = e^{-x}$, uzimajući da je $e^{-1} = 0.367879$, $e^{-2} = 0.135335$, $e^{-3} = 0.0497871$. Treba interpolirati funkciju polinomom drugog stupnja metodom najmanjih kvadrata i pomoću njega približno naći $f(5.5)$. [39]

Rješenje:

$$p(x) = a + bx + cx^2$$

je polinom koji ćemo koristiti i on interpolira sljedeće podatke.

Tablica 18: Zadatak-metoda najmanjih kvadrata

x	1	2	3
$f(x)$	0.367879	0.135335	0.0497871

Sada formulu

$$S(\alpha, \beta) = \sum_{k=1}^{n}[f(x_k) - \alpha x_k - \beta]^2$$

napišemo u obliku

$$S(a, b, c) = \sum_{k=1}^{3}[f(x_k) - cx^2 - bx_k - a]^2.$$

Ova funkcija ovisi o tri varijable, pa je nužan uvjet za ekstrem

$$\frac{\partial S(a,b,c)}{\partial a} = -2\sum_{k=1}^{3}[f(x_k) - cx^2 - bx_k - a] = 0$$

$$\frac{\partial S(a,b,c)}{\partial b} = -2\sum_{k=1}^{3}[f(x_k) - cx^2 - bx_k - a]x_k = 0$$

$$\frac{\partial S(a,b,c)}{\partial c} = -2\sum_{k=1}^{3}[f(x_k) - cx^2 - bx_k - a]x_k^2 = 0.$$

Kad se uvrste poznate veličine i sredi izraz, dobije se

$$\begin{cases} 6a + 12b + 28c = 1.106 \\ 12a + 28b + 72c = 1.57582 \\ 28a + 72b + 196c = 2.71461 \end{cases}.$$

Sređivanjem sustava dobije se rješenje $a = 0.74742,\ b = -0.453038,\ c = 0.073498$, pa je tako traženi polinom

$$p(x) = 0.74742 - 0.453038x + 0.073498x^2,$$

a tražena vrijednost $f(5.5) = 0.4790255.$

Popis literature

[1]: Vuković, D., „Numerička analiza-interpolacija i ekstrapolacija“, 2017. Dostupno: https://kupdf.com/download/numericka-analiza_59e24ead08bbc5277fe65498_pdf [pristupano 30.10.2017.]

[2]: Bronštejn, Semendjajev, Musiol, Mühlig, *Matematički priručnik,* Golden marketing – Tehnička knjiga, Zagreb, 2004.

[3]: Lesić, M., „Aproksimacija i interpolacija funkcija“, 2017. Dostupno: https://zir.nsk.hr/islandora/object/vuka%3A455 [pristupano 30.10.2017.]

[4]: Žitnik, T., „Neville-ov postupak interpolacije“, 2001. Dostupno: web.zpr.fer.hr/zpm13c2/zitnik/neville.doc [pristupano 30.10.2017.]

[5]: Ujević, N., „Uvod u numeričku matematiku“, 2004. Dostupno: https://www.yumpu.com/xx/document/view/24243491/uvod-u-numeriecku-matematiku-fakultet-prirodoslovno-/7 [pristupano 30.10.2017.]

[6]: Slanivuk, T., „Aproksimacija i interpolacija“, 2014. Dostupno: https://www.slideserve.com/argyle/aproksimacija-i- interpolacija [pristupano 31.10.2017.]

[7]: Singer, S., „Numerička matematika 4. predavanje“, 2017. Dostupno: https://web.math.pmf.unizg.hr/~singer/num_mat/NM_1617/04.pdf [pristupano 31.10.2017.]

[8]: Jukić, Scitovski, „Matematika 1“, 1998. Dostupno: https://www.mathos.unios.hr/integralni/Jukic_Scitovski.pdf [pristupano 3.11.2017]

[9]: Brückler, F. M., „Redovi potencija“, (bez dat.) Dostupno: http://prelog.chem.pmf.hr/~fmbruckler/mat2-pred14.pdf [pristupano 3.11.2017.]

[10]: Beros, „Uvod u numeričku matematiku vježbe“, 2004. Dostupno: http://degiorgi.math.hr/forum/download.php?id=721 [pristupano 5.11.2017.]

[11]: „mat_9_2“, 2007. Dostupno: https://www.fsb.unizg.hr/mat-4/OldWeb/mat_9_2.pdf [pristupano 5.11.2017.]

[12]: „Interpolacija Lagrange-ovim polinomom“, 2008. Dostupno: http://161.53.18.5/zpr/Portals/0/Predmeti/PNP/08-InterpolacijaLagrangePolinomom.pdf [pristupano 5.11.2017.]

[13]: Beros, „Uvod u numeričku matematiku vježbe“, 2004. Dostupno: http://degiorgi.math.hr/forum/download.php?id=721 [pristupano 5.11.2017.]

[14]: Kovačić, B., „Matematički alati u elektrotehnici“, 2013. Dostupno: http://bkovacic.weebly.com/uploads/7/4/0/7/7407552/ma_-_skripta.pdf [pristupano 5.11.2017.]

[15]: „Interpolacija“, 2006. Dostupno: http://etf.sastavisam.com/IR2NUMDIS/num/vezbe1_okt2006.pdf [pristupano 6.11.2017.]

[16]: Krstinić, D., „Primijenjena matematika“, (bez dat.) Dostupno: marjan.fesb.hr/~borka/files/prmat/prmat_Bisekcija_Interpolacija.ppt [pristupano 6.11.2017.]

[17]: „Interpolacija i aproksimacija funkcija“, 2017. Dostupno: https://element.hr/artikli/file/1314 [pristupano 8.11.2017.]

[18 Singer, S., „Numerička matematika 5. predavanje“, 2017. Dostupno: https://web.math.pmf.unizg.hr/~singer/num_mat/NM_1617/05.pdf [pristupano 9.11.2017.]

[19]: Klaričić Bakula, M., „Uvod u Numeričku matematiku“, 2009. Dostupno: http://mapmf.pmfst.unist.hr/~milica/Numericka_matematika/Folije_za_predavanja/UNM_Interp.pdf [pristupano 11.11.2017.]

[20]: Milovanović, Kovačević, Spalević, „Numerička matematika - zbirka rešenih problema“, 2002. Dostupno: https://ar.scribd.com/document/98278997/ZbirkaNumerickaAnaliza-1 [pristupano 11.11.2017.]

[21]: Omorjan, R., „Interpolacija“, 2003. Dostupno: http://www.tf.uns.ac.rs/~omorr/radovan_omorjan_003_nm/02Interpolacija.pdf [pristupano 12.11.2017.]

[22]: Drmač, Hari, Marušić, Rogina, Singer, Singer, „Numerička analiza“, 2003. Dostupno: https://web.math.pmf.unizg.hr/~singer/num_mat/num_anal.pdf [pristupano 15.11.2017.]

[23]: Omorjan, R., 2003. Dostupno: http://www.tf.uns.ac.rs/~omorr/radovan_omorjan_003_hip/Dodatak%20B.pdf [pristupano 15.11.2017.]

[24]: „Numeričko diferenciranje i integriranje“ 2014. Dostupno: https://www.pmf.unizg.hr/_download/repository/PREDAVANJE4.pdf [pristupano 15.11.2017.]

[25]: „Interpolacija krivih: Interpolacioni polinomi“, 2013. Dostupno: http://imft.ftn.uns.ac.rs/~natasa/old/uploads/Main/Interpolacija1.pdf [pristupano 18.11.2017.]

[26]: Volmut, V., „Ortogonalni polinomi“, 2016. Dostupno: https://zir.nsk.hr/islandora/object/mathos%3A87 [pristupano 18.11.2017.]

[27]:Grubišić, L., „Numerička matematika Predavanja 09/10“, 2010., Dostupno: https://web.math.pmf.unizg.hr/~luka/Nastava/?download=Pred_0910_trece.pdf [pristupano 18.11.2017.]

[28]: Radunović, D. P., „Numeričke metode“, 2003. Dostupno: http://elibrary.matf.bg.ac.rs/bitstream/handle/123456789/2936/NUMET_knjiga.pdf?sequence=1 [pristupano 18.11.2017.]

[29]: Drmač, Hari, Marušić, Rogina, Singer, Singer, „Numerička analiza- skraćena skripta“, 2008. Dostupno: https://web.math.pmf.unizg.hr/~singer/num_mat/num_mat1.pdf [pristupano 22.11.2017.]

[30]: „Fourierov red po ortogonalnim sustavima“, 2015. Dostupno: http://manualzz.com/doc/11573919/p07--246-46-kib- [pristupano 25.11.2017.]

[31]: Vranješ Markić, L., „Matematičke metode fizike 1“, 2009. Dostupno: https://pt.scribd.com/doc/126346993/MATEMATIKA-4 [pristupano 29.11.2017.]

[32]: „336 B-splajn“ (bez dat.) Dostupno: https://www.fsb.unizg.hr/geometrija.broda/300/330/gb336.htm [pristupano 7.12.2017.]

[33]: Carpenter, K. H., „B-splines“, 2001. Dostupno: www.ece.k-state.edu//people/faculty/carpenter/documents/bsplpdf.pdf [pristupano 7.12.2017.]

[34]: „B-krivulje ili B-splajnovi - B-spline Curves“, 2013. Dostupno: http://www.lecad.unze.ba/nastava/cadteh/CAD5Krivulje/CAD5-7B-splajnovi.pdf [pristupano 7.12.2017.]

[35]: Marjan, „Matematičko modeliranje forme trupa broda“, 2008. Dostupno: http://marjan.fesb.hr/~bblag/publications/books/matematicko_modeliranje_forme_trupa_broda.pdf [pristupano 7.12.2017.]

[36]: „Spline, Bezier, B-spline“, 2016. Dostupno: www.mmumullana.org/downloads/files/n547455c832229.pdf [pristupano 7.12.2017.]

[37]: Malaek, S. M., „Curve Modeling - B-spline Curves“, (bez dat.) Dostupno: http://slideplayer.com/slide/6930421/ [pristupano 7.12.2017.]

[38]: „Lagrangeov polinom“, 2008. Dostupno: http://m.bug.hr/forum/topic/programiranje/lagrangeov-polinom/9642 [pristupano 12.12.2017.]

[39]: Drakos, N. „Metoda najmanjih kvadrata“, 2001. Dostupno: http://www.grad.hr/nastava/matematika/mat3/node153.html [pristupano 15.12.2017.]

[40]: Interpolacija (lijevo) i aproksimacija (desno) [Slika] (bez dat.) Preuzeto: 30.10.2017. s http://www.grad.hr/nastava/matematika/mat3/node128.html

[41]: Funkcija sin x i Taylorovi polinomi [Slika] (bez dat.) Preuzeto: 3.11.2017. s https://sh.wikipedia.org/wiki/Tejlorov_polinom

[42]: Lagrangeov interpolacijski polinom u MATLAB-u [Slika] (bez dat.) Preuzeto: 5.11.2017. s [14]

[43]: Prvih nekoliko Hermiteovih polinoma [Slika] (bez dat.) Preuzeto: 18.11.2017. s www.mathepedia.de/html/a_analysis/8_dim1/f_spezfunc/c_spezpolynome/Hermite.aspx?w=500&h=400

[44]: Interpolacijski polinomi stupnja 1-6 na ekvidistantnoj mreži [Slika] (bez dat.) Preuzeto: 22.11.2017. s [29]

[45]: Greške interpolacijskih polinoma stupnja 1-6 na ekvidistantnoj mreži [Slika] (bez dat.) Preuzeto: 22.11.2017 s [29]

[46]: Interpolacijski polinomi stupnja 1-6 na Čebiševljevoj mreži [Slika] (bez dat.) Preuzeto: 22.11.2017 s [29]

[47]: Greške interpolacijskih polinoma stupnja 1-6 na Čebiševljevoj mreži [Slika] (bez dat.) Preuzeto: 22.11.2017 s [29]

[48]: Čebiševljevi polinomi prve vrste [Slika] (bez dat.) Preuzeto: 25.11.2017. s [18]

[49]: Čebiševljevi polinomi druge vrste [Slika] (bez dat.) Preuzeto: 27.11.2017. s https://en.wikipedia.org/wiki/Chebyshev_polynomials

[50]: Interpolacija funkcije polinomom [Slika] (bez dat.) Preuzeto: 29.11.2017. s [31]

[51]: Interpolacija funkcije spline-om [Slika] (bez dat.) Preuzeto: 29.11.2017. s [31]

[52]: Prikaz linearnih spline-ova za funkciju f(x) [Slika] Autorski rad: 5.12.2017.

[53]: Pomicanje točke kod B-spline-a [Slika] (bez dat.) Preuzeto: 7.12.2017. s [36]

[54]: Vrste B-spline-ova prema strukturi [Slika] (bez dat.) Preuzeto: 7.12.2017. s [37]

[55]: Konstruiranje B-spline-a iz otvorenog u zatvoreni [Slika] (bez dat.) Preuzeto: 7.12.2017. s [37]

[56]: Rezultati algoritma linearnog spline-a za funkciju sin x [Slika] Autorski rad: 11.12.2017.

[57]: Rezultati algoritma linearnog spline-a za funkciju 1/(x^2+1) [Slika] Autorski rad: 11.12.2017.

[58]: Rezultati Lagrangeovog algoritma [Slika] Autorski rad: 12.12.2017.

Popis slika

Popis tablica

Prilozi

1. Implementacija algoritma linearnog spline-a
2. Implementacija algoritma Lagrangeovog oblika interpolacijskog polinoma