

Univerzitet u Beogradu  
Matematički fakultet

**KONSTRUKCIJA INTERPOLACIONIH  
TALASIĆA ZA REŠAVANJE  
DIFERENCIJALNIH JEDNAČINA**  
*-Master rad-*

Student: Milan Dotlić  
Mentor: prof. dr. Desanka Radunović

Beograd, 2010.

# Sadržaj

<b>1</b>	<b>Uvod</b>	<b>4</b>
<b>2</b>	<b>Talasići</b>	<b>7</b>
2.1	Multirezolucija . . . . .	7
2.2	Druga generacija talasića . . . . .	10
2.3	Interpolacioni talasići . . . . .	13
2.4	Lifting šema . . . . .	18
<b>3</b>	<b>Primena na diferencijalne jednačine</b>	<b>23</b>
3.1	Numeričko rešavanje diferencijalnih jednačina . . . . .	23
3.2	Aproksimacija izvoda . . . . .	27
3.3	Numerički rezultati . . . . .	29
<b>4</b>	<b>Zaključak</b>	<b>43</b>
	<b>Literatura</b>	<b>44</b>

# Slike

2.1	Interpolacione funkcije skaliranja za $N=2,4,6,8$ . . . . .	14
2.2	Interpolacione funkcije skaliranja za $N=4$ kada na njih utiče granica . . . . .	15
2.3	$f(x) = \cos(80\pi x) \exp^{-64x^2}$ . . . . .	17
2.4	Distribucija koeficijenata $d_{j,k}$ (prikazani samo koeficijenti za koje je $ d_{j,k}  > 10^{-3}$ ) . . . . .	18
2.5	Distribucija koeficijenata $d_{j,k}$ (prikazani samo koeficijenti za koje je $ d_{j,k}  > 10^{-3}$ ) . . . . .	22
3.1	Analitičko rešenje primera 1. za $\theta = 0.01(-)$ , $\theta = 0.0001(-)$ , $\theta = 0.0000001(-)$ . . . . .	30
3.2	Rešenja i raspored tačaka kolokacije pri korišćenju prvog algoritma na prvom primeru za $\epsilon = 10^{-3}$ i redom $\theta = 10^{-2}, 10^{-4}, 10^{-7}$ . . . . .	32
3.3	Rešenja i raspored tačaka kolokacije pri korišćenju drugog algoritma na prvom primeru za $\epsilon = 10^{-3}$ i redom $\theta = 10^{-2}, 10^{-4}, 10^{-7}$ . . . . .	33
3.4	Analitičko rešenje primera 2. pri izboru $\theta = 0.1(-)$ , $\theta = 0.05(-)$ , $\theta = 0.01(-)$ . . . . .	35
3.5	Multirezolucijski nivoi pri loše izabranom $J$ . . . . .	36
3.6	Rešenje i raspored tačaka pri korišćenju prvog algoritma na drugom primeru za $\epsilon = 10^{-5}$ , i redom $\theta = 0.1, 0.05, 0.01$ . . . . .	38
3.7	Rešenje i raspored tačaka pri korišćenju drugog algoritma na drugom primeru za $\epsilon = 10^{-5}$ , i redom $\theta = 0.1, 0.05, 0.01$ . . . . .	39
3.8	Analitičko rešenje primera 3. ukoliko je $\theta = 0.1(-)$ , $\theta = 0.01(-)$ , $\theta = 0.0001(-)$ . . . . .	40
3.9	Rešenja i raspored tačaka kolokacije pri korišćenju prvog algoritma na trećem primeru za $\epsilon = 0.00001$ i redom $\theta=0.1 (-)$ , $\theta=0.01(-)$ , $\theta=0.0001(-)$ . . . . .	41
3.10	Rešenja i raspored tačaka kolokacije pri korišćenju drugog algoritma na trećem primeru za $\epsilon = 0.00001$ i redom $\theta=0.1 (-)$ , $\theta=0.01(-)$ , $\theta=0.0001(-)$ . . . . .	42

# 1

## Uvod

U ovom radu biće predstavljeno rešavanje diferencijalnih jednačina (DJ) korišćenjem talasića (wavelets). Obične i parcijalne diferencijalne jednačine predstavljaju matematičke modele velikog broja fizičkih, hemijskih, bioloških i ekonomskih pojava (npr. širenje zaraza, vremenska prognoza, kretanja na berzi). Vrlo često je nemoguće ili nepraktično rešavanje ovakvih jednačina. Stoga se koriste numeričke aproksimacije umesto analitičkog rešavanja.

Postoji nekoliko metoda baziranih na talasićima koje grubo možemo podeliti na vejevlet-Galerkinove<sup>1</sup> ([1], [3]) i vejevlet-kolokacijske metode ([4], [19], [20], [22], [5], [6]). U ovom radu će biti reči o vejevlet-kolokacijskim metodama. U opštem slučaju ideja metode kolokacije je da se izaberu bazisne funkcije i mreža (tačke kolokacije), tako da u tačkama kolokacije možemo predstaviti rešenje pomoću bazisnih funkcija.

Posebna pažnja biće posvećena singularno-perturbovanim problemima. Singularno-perturbovani problemi su diferencijalne jednačine sa parametrom čije rešenje ispoljava neuniformno ponašanje kada parametar teži ka nekoj graničnoj vrednosti. Neuniformnost ovih problema ogleda se u pojavi slojeva u kojima se rešenje naglo menja kada parametar teži nuli. Veliku poteškoću u rešavanju standardnim metodama numeričke analize predstavljaju upravo problemi čija su rešenja glatka u najvećem delu domena, sa malim oblastima gde dolazi do velikih i brzih promena. Numeričko rešavanje takvih problema na ravnomernoj mreži metodom kolokacije je nepraktično. Nepraktičnost se ispoljava u činjenici da je velika rezolucija potrebna samo u regionima gde dolazi do tih promena, tj. poželjno je da u oblastima gde su velike varijacije ima više tačaka. Shodno tome potrebna je i adaptacija mreže.

U vejevlet-kolokacijskim metodama svaki talasić jedinstveno je povezan sa jednom tačkom kolokacije, tako da se adaptacija mreže izvodi analizom

---

<sup>1</sup>Boris Grigoryevich Galerkin

koeficijenata talasića. Drugim rečima, u svakom trenutku vremena mreža se sastoji od tačaka koje odgovaraju talasićima čiji su koeficijenti veći od zadatog parametra koji kontroliše tačnost. U ovoj metodi se izvodi mogu aproksimirati na nekoliko načina, a u ovom radu se koriste operatori podeljenih razlika.

Pre predstavljanja metode, osvrnimo se na kratak istorijat talasića. Termin talasić (eng. wavelet) prvi put je upotrebljen osamdesetih godina XX veka, od strane Grosmana<sup>2</sup> i Morleta<sup>3</sup>. Takav naziv je dat upravo zbog talasne prirode funkcija koje se koriste, a deminutiv se koristi zbog njihove ograničenosti. Sama ideja reprezentacije funkcije dosta je starija od talasića. Ideja potiče od Furijea<sup>4</sup>, koji je 1807. godine izneo tezu da je svaka  $2\pi$  periodična integrabilna funkcija suma svog Furijevog reda

$$\frac{a_0}{2} + \sum_k (a_k \cos kx + b_k \sin kx),$$

pri odgovarajućim vrednostima koeficijenata  $a_k, b_k$ .

Har<sup>5</sup> je razvio ovu ideju dalje. Tražio je ortonormirani sistem funkcija na intervalu  $[0,1]$  takav da, za bilo koju funkciju  $f(x)$  neprekidnu na tom intervalu, red

$$(f, h_0)h_0(x) + (f, h_1)h_1(x) + \dots + (f, h_n)h_n(x) + \dots,$$

sa skalarnim proizvodom

$$(f, g) = \int_0^1 f(x)\overline{g(x)}dx,$$

uniformno konvergira ka  $f(x)$  na intervalu  $[0,1]$ . Ovaj problem ima beskonačno mnogo rešenja. Har je dao najjednostavnije rešenje koje vodi do talasića. Za bazisnu funkciju  $h_n(x)$  izabrao je karakterističnu funkciju diadskog intervala  $I_n = [2^{-j}k, 2^{-j}(k+1)]$ ,  $n = 2^j + k$ , koja je jednaka jedan na tom intervalu i nula inače. Upravo ovo je dobro poznata aproksimacija  $L_2$  integrabilne funkcije deo po deo konstantom, pri čemu su  $(f, h_n)$  srednje vrednosti funkcije  $f(x)$  na odgovarajućim diadskim intervalima. Velika prednost u odnosu na Furijeovu analizu, koja se zasniva na jednom skupu funkcija, jeste mogućnost reprezentacije talasićima po beskonačno mnogo bazisa.

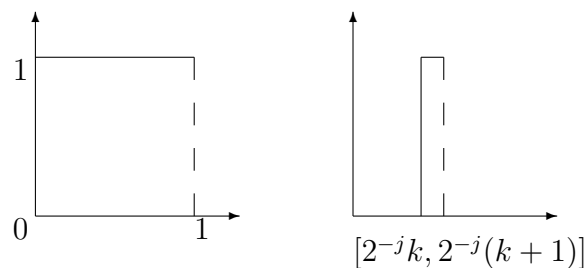
---

<sup>2</sup>Alex Grossmann

<sup>3</sup>Jean Morlet

<sup>4</sup>Jean-Baptiste Joseph Fourier

<sup>5</sup>Alfréd Haar



Harova četvrtka

Do ponovnog “otkrivanja” talasića dolazi osamdesetih godina XX veka. Tada je Harova teorija upotpunjena radovima Ingrid Dobešis<sup>6</sup>, koja je konstruisala različite familije ortonormiranih bazisa talasića. Krajem XX i početkom XXI veka dolazi do ogromnog napretka u ovoj oblasti numeričke analize.

Prednost talasića predstavlja ogromno bogatstvo različitih bazisa, tako da u zavisnosti od potrebe mogu se konstruisati odgovarajući bazisi.

Ovaj rad je podeljen u četiri glave. Druga glava je posvećena talasićima, tu je objašnjena klasična multirezolucijska analiza, druga generacija talasića i interpolacioni talasići kao posebno pogodni za modelovanje diferencijalnih jednačina. Takođe, u drugoj glavi je predstavljena i lifting šema kao način za konstrukciju talasića druge generacije. Treća glava posvećena je primeni talasića u rešavanju diferencijalnih jednačina. U istoj glavi u delu numeričko rešavanje diferencijalnih jednačina predstavljeni su algoritmi za rešavanje DJ. Dalje, ova glava objašnjava aproksimaciju izvoda, a potom predstavlja i dobijene rezultate.

---

<sup>6</sup>Ingrid Daubechies

## 2

# Talasići

## 2.1 Multirezolucija

Pre definisanja multirezolucije, podsetimo se definicije Lebegovih prostora i Risovog bazisa koje su neophodne za dalji rad.

Funkcija  $f$  pripada Lebegovom prostoru  $L_p(\mathbf{R})$ ,  $1 \leq p < \infty$ , ako važi:

$$\|f\|_p = \left( \int_{-\infty}^{\infty} |f(x)|^p dx \right)^{1/p} < \infty,$$

odnosno  $L_\infty(\mathbf{R})$  ukoliko važi

$$\|f\|_\infty = \text{esssup}_{x \in \mathbf{R}} |f(x)| < \infty.$$

Svaki Lebegov prostor je i Banahov prostor. Njihovi elementi su klase ekvivalencije funkcija koje se podudaraju skoro svuda. Prostor  $L_2(\mathbf{R})$  je Hilbertov prostor gde je skalarni proizvod funkcija  $f$  i  $g$  zadat sa:

$$(f, g) = \int_{-\infty}^{\infty} f(x) \overline{g(x)} dx.$$

Dve funkcije su ortogonalne ukoliko je njihov skalarni proizvod jednak nuli.

**Definicija 1.** *Risovim bazisom nazivamo prebrojiv skup elemenata  $\{g_k\}$  Hilbertovog prostora koji zadovoljava uslov da se svaki element  $f$  tog prostora može jednoznačno predstaviti u obliku sume  $f = \sum_k c_k g_k(x)$ , pri čemu postoje pozitivne konstante  $A$  i  $B$  ( $0 < A \leq B < \infty$ ) takve da je:*

$$A \|f\|^2 \leq \sum_k |c_k|^2 \leq B \|f\|^2.$$

U konačno dimenzionom prostoru svaki bazis je Risov bazis. Ortonormirani bazis je Risov bazis sa konstantama  $A = B = 1$ . Bazis  $1, x, x^2, \dots$  nije Risov bazis u  $L_2(0, 1)$ , jer je konstanta  $A = 0$ .

**Definicija 2.** *Multirezolucijska analiza je dekompozicija Hilbertovog prostora  $L_2(\mathbf{R})$  na niz zatvorenih potprostora  $\{V_j\}_{j \in \mathbf{Z}}$  takvih da je:*

$$V_j \subset V_{j+1}, \quad \forall j \in \mathbf{Z} \quad (2.1)$$

$$\bigcap_{j \in \mathbf{Z}} V_j = \{0\}, \quad \overline{\bigcup_{j \in \mathbf{Z}} V_j} = L_2(\mathbf{R}) \quad (2.2)$$

$$\forall f \in L_2(\mathbf{R}) \quad i \quad \forall j \in \mathbf{Z}, \quad f(x) \in V_j \Leftrightarrow f(2x) \in V_{j+1} \quad (2.3)$$

$$\forall f \in L_2(\mathbf{R}) \quad i \quad \forall k \in \mathbf{Z}, \quad f(x) \in V_0 \Leftrightarrow f(x - k) \in V_0 \quad (2.4)$$

$$\exists \varphi \in V_0 \text{ tako da je } \{\varphi(x - k)\}_{k \in \mathbf{Z}} \text{ Risov bazis potprostora } V_0. \quad (2.5)$$

Svaki aproksimacioni prostor  $V_j$ ,  $j \in \mathbf{Z}$ , je skalirana verzija prostora  $V_0$ , dobijena skaliranjem (za faktor  $2^{-j}$ ), tj. širenjem ili skupljanjem (zavisno od znaka  $j$ ) prostora  $V_0$ . Osnovni prostor  $V_0$  generisan je jednom funkcijom  $\varphi \in L_2(\mathbf{R})$ , koja se naziva funkcija skaliranja, s obzirom da bazis prostora  $V_0$ , prema (2.5), obrazuju funkcija  $\varphi(x)$  i njene translacije  $\varphi(x - k)$ ,  $k \in \mathbf{Z}$ . Ako funkciju  $\varphi(x)$  skaliranu  $j$  puta i transliranu za  $k$  označimo sa

$$\varphi_{j,k}(x) = 2^{j/2} \varphi(2^j x - k) \quad j, k \in \mathbf{Z},$$

Risov bazis prostora  $V_j$  je skup funkcija  $\{\varphi_{j,k}\}$ ,  $k \in \mathbf{Z}$ .

**Teorema 1.** *Ako je skup funkcija  $\{\varphi(x - k) | k \in \mathbf{Z}\}$  Risov bazis prostora  $V_0$ , onda je svaki od skupova  $\{\varphi_{j,k} | k \in \mathbf{Z}\}$  Risov bazis prostora  $V_j$ ,  $j \in \mathbf{Z}$  gde je  $\varphi_{j,k}(x) = 2^{j/2} \varphi(2^j x - k)$   $j, k \in \mathbf{Z}$ .*

*Dokaz:* Neka je  $f(x) \in V_j$  proizvoljna funkcija. Tada je (na osnovu (2.3))  $f(2^{-j}x) \in V_0$ . Ako je  $\{\varphi(x - k) | k \in \mathbf{Z}\}$  Risov bazis prostora  $V_0$ , važi:

$$(\forall x \in \mathbf{R}) f(2^{-j}x) = \sum_n c_n \varphi(x - n),$$

pri čemu postoje konstante  $A$  i  $B$  ( $0 < A \leq B < \infty$ ) tako da važi:

$$A \|f\|^2 \leq \sum_n |c_n|^2 \leq B \|f\|^2.$$

Stoga je:

$$f(x) = f(2^{-j}2^j x) = \sum_n c_n \varphi(2^j x - n),$$

pa je skup  $\{\varphi_{j,k} | k \in Z\}$  zaista bazis prostora  $V_j$ , a uz to je i Risov bazis zbog:

$$2^j A \|f\|^2 \leq \sum_n |c_n|^2 \leq 2^j B \|f\|^2.$$

□

**Teorema 2.** *Svaki od prostora  $V_j, j \in Z$ , je invarijantan u odnosu na translacije, odnosno*

$$f(x) \in V_j \Leftrightarrow (\forall k \in Z) f(x - k) \in V_j, j \in Z.$$

*Dokaz:* Direktno sledi iz (2.4). □

Prelazeći sa prostora  $V_j$  na prostor  $V_{j-1}$  dolazi do gubljenja detalja. Kako važi  $V_{j-1} \subset V_j$ , izgubljeni detalji su sačuvani u ortogonalnom komplementu potprostora  $V_{j-1}$  u odnosu na prostor  $V_j$ . Ovaj ortogonalni komplement ćemo označavati sa  $W_{j-1}$  i zvati prostor talasića

$$V_j = V_{j-1} \oplus W_{j-1}.$$

Iz poslednje relacije sledi da su prostori talasića razlike aproksimacionih prostora  $V_j$ . Dalje, imamo da je:

$$V_j = V_{j-1} \oplus W_{j-1} \quad i \quad V_{j-1} = V_{j-2} \oplus W_{j-2},$$

odakle sledi

$$V_j = W_{j-1} \oplus W_{j-2} \oplus V_{j-2}.$$

Daljim razlaganjem po istom principu, koristeći uslov (2.2), dolazimo do razlaganja prostora  $L_2(\mathbf{R})$ . Kada  $j \rightarrow \infty$  imamo dekompoziciju,

$$L_2(\mathbf{R}) = V_J \oplus \sum_{j=J}^{\infty} W_j,$$

odnosno kada  $J \rightarrow -\infty$ ,

$$L_2(\mathbf{R}) = \sum_{j=-\infty}^{\infty} W_j.$$

Slično aproksimacionim prostorima  $V_j$ , prostori talasića  $W_j$  generisani su skaliranjem i diadskim translacijama druge funkcije  $\psi(x) \in L_2(\mathbf{R})$ , koja se naziva osnovni talasić. U tom smislu je

$$W_j = \overline{\{\psi_{j,k} \in L_2(\mathbf{R}) | k \in Z\}}, \quad \psi_{j,k}(x) = 2^{j/2} \psi(2^j x - k), \quad j \in Z.$$

Ovakvo razlaganje prostora funkcija  $L_2(\mathbf{R})$  na multirezolucijske potprostore, daje mogućnost razlaganja proizvoljne funkcije  $f(x) \in L_2(\mathbf{R})$ , tako da postoji komponenta funkcije  $f(x)$  u svakom od ovih potprostora.

$$f(x) = \sum_{k \in \mathbf{Z}} \underbrace{c_{J,k} \varphi_{J,k}(x)}_{\in V_J} + \sum_{j=J}^{\infty} \sum_{k \in \mathbf{Z}} \underbrace{d_{j,k} \psi_{j,k}(x)}_{\in W_j}. \quad (2.6)$$

Način izračunavanja koeficijenata  $c_{j,k}$  i  $d_{j,k}$  biće kasnije objašnjen.

Uporedo sa ovim uvedimo i pojam biortogonalnih talasića. Umesto jednog, konstruišu se dva niza multirezolucijskih prostora

$$V_j = V_{j-1} \oplus W_{j-1}, \quad \tilde{V}_j = \tilde{V}_{j-1} \oplus \tilde{W}_{j-1}.$$

Sume prostora su direktne, međutim u opštem slučaju nisu ortogonalne. Ortogonalnost, tj. biortogonalnost postoji između različitih multirezolucijskih prostora:

$$V_j \perp \tilde{W}_j, \quad W_j \perp \tilde{V}_j.$$

I u slučaju ovako definisane multirezolucije prostori  $\tilde{V}_j$  imaju Risov bazis zadat funkcijama  $\tilde{\varphi}_{j,k}$ , a prostori  $\tilde{W}_j$  imaju Risov bazis zadat funkcijama  $\tilde{\psi}_{j,k}$ . Za biortogonalne funkcije skaliranja i talasiće važe relacije:

$$\begin{aligned} (\tilde{\varphi}_{j,k}, \varphi_{j',k'}) &= \delta_{j,j'} \delta_{k,k'}, \\ (\tilde{\psi}_{j,m}, \psi_{j',m'}) &= \delta_{j,j'} \delta_{m,m'}, \\ (\tilde{\varphi}_{j,k}, \psi_{j,m}) &= 0, \\ (\tilde{\psi}_{j,m}, \varphi_{j,k}) &= 0. \end{aligned}$$

## 2.2 Druga generacija talasića

Zbog translacije i dilatacije javlja se problem aproksimacije funkcije na konačnom intervalu standardnom transformacijom talasićima. Stoga ćemo standardnu ideju multirezolucije oslabiti ne zahtevajući translaciju i dilataciju. Ovo će dovesti do druge generacije talasića i metode za njihovu konstrukciju poznatiju kao lifting šema ([13],[14]). Prednost druge generacije talasića sastoji se u tome što konkretan izbor bazisa talasića lako možemo proširiti na potpuno novu klasu talasića druge generacije, ostavljajući slobodu da se izaberu talasići u zavisnosti od primene. U skladu sa prethodnim razmatranjima definišimo multirezoluciju druge generacije.

**Definicija 3.** *Multirezolucijska analiza druge generacije je dekompozicija Hilbertovog prostora  $L_2(\mathbf{R})$  na niz zatvorenih potprostora  $\{V_j\}_{j \in Z}$  takvih da je:*

$$V_j \subset V_{j+1}, \quad \forall j \in Z \quad (2.7)$$

$$\bigcap_{j \in Z} V_j = \{0\}, \quad \overline{\bigcup_{j \in Z} V_j} = L_2(\mathbf{R}) \quad (2.8)$$

$$\forall j \in Z, V_j \text{ ima Risov bazis zadat sa } \{\varphi_{j,k} | k \in Z\}. \quad (2.9)$$

Primetimo da u poredjenju sa definicijom 2. ovde nedostaju uslovi (2.3) i (2.4). Dualna multirezolucija se definiše na sličan način. Sastoji se od prostora  $\tilde{V}_j$ , generisanih dualnim funkcijama skaliranja  $\tilde{\varphi}_{j,k}$  koje su ortogonalne na funkcije skaliranja primala  $\varphi_{j,k}$ .

Kako  $\varphi_{j,k}$  pripada  $V_j$ , pa samim tim i  $V_{j+1}$ , to funkciju skaliranja možemo predstaviti pomoću bazisnih funkcija prostora  $V_{j+1}$ :

$$\varphi_{j,k} = \sum_l h_{j,k,l} \varphi_{j+1,l}.$$

Talasiće uvodimo slično kao kod obične multirezolucije, kao bazis funkcija za prostor  $W_j$ . S obzirom da talasići  $\psi_{j,k}$  pripadaju  $W_j \subset V_{j+1}$ , njih možemo predstaviti slično kao i funkcije skaliranja:

$$\psi_{j,k} = \sum_l g_{j,k,l} \varphi_{j+1,l}.$$

Kako  $\varphi_{j+1,k} \in V_j \oplus W_j$ ,  $\varphi_{j+1,k}$  može se predstaviti kao linearna kombinacija bazisa prostora  $V_j$  i  $W_j$ :

$$\varphi_{j+1,k} = \sum_l \tilde{h}_{j,k,l} \varphi_{j,l} + \sum_m \tilde{g}_{j,k,m} \psi_{j,m}.$$

Pokažimo sada kako funkcioniše transformacija talasićima druge generacije. Ukoliko su dati koeficijenti funkcije skaliranja  $c_{j+1,k}$  na nivou  $j+1$ , kombinujući poslednje tri jednakosti i jednačinu (2.6) mogu se naći koeficijenti talasića  $d_{j,k}$  i funkcije skaliranja  $c_{j,k}$  na nivou  $j$  (više o izvodjenju u [9],[13],[14]):

$$d_{j,k} = \sum_l \tilde{g}_{j,k,l} c_{j+1,l}, \quad (2.10)$$

$$c_{j,k} = \sum_l \tilde{h}_{j,k,l} c_{j+1,l}. \quad (2.11)$$

Inverzna transformacija je data sa

$$c_{j+1,k} = \sum_m h_{j,k,m} c_{j,m} + \sum_l g_{j,k,l} d_{j,l}. \quad (2.12)$$

Za koeficijente (filtre)  $h, \tilde{h}, g, \tilde{g}$  zbog biortogonalnosti važi ([13],[14]):

$$\begin{aligned} \sum_l g_{j,m,l} \tilde{g}_{j,m',l} &= \delta_{m,m'}, \\ \sum_l h_{j,k,l} \tilde{h}_{j,k',l} &= \delta_{k,k'}, \\ \sum_l h_{j,k,l} \tilde{g}_{j,m,l} &= 0, \\ \sum_l g_{j,m,l} \tilde{h}_{j,k,l} &= 0. \end{aligned} \quad (2.13)$$

Pre matricnog zapisa uvedimo oznake  $K(j), K(j+1)$  i  $M(j)$ .  $K(j)$ , odnosno  $K(j+1)$  predstavlja skup indeksa na  $j$  nivou, odnosno  $j+1$  nivou, a  $M(j)$  predstavlja  $M(j) = K(j+1) \setminus K(j)$ .

Neka su  $p, q, r$  takvi da:

$$p = \{p_l | l \in K(j+1)\},$$

$p \in \ell^2(K(j+1))$ . Slično  $q \in \ell^2(K(j))$  i  $r \in \ell^2(M(j))$ .

Uvedimo sada operatore  $H_j$  i  $G_j$ , na sledeći način:

$H_j : \ell^2(K(j+1)) \longrightarrow \ell^2(K(j))$ , gde  $q = H_j p$  ima elemente:

$$q_k = \sum_{l \in K(j+1)} h_{j,k,l} p_l,$$

$G_j : \ell^2(K(j+1)) \rightarrow \ell^2(M(j))$ , gde  $r = G_j p$  ima elemente:

$$r_m = \sum_{l \in K(j+1)} g_{j,m,l} p_l.$$

Zapisano u matricnom obliku:

$$H_j = \begin{pmatrix} h_{j,0,0} & h_{j,0,1} & \dots & h_{j,0,l} & \dots \\ & & \vdots & & \\ h_{j,k,0} & h_{j,k,1} & \dots & h_{j,k,l} & \dots \\ & & \vdots & & \end{pmatrix}$$

$$G_j = \begin{pmatrix} g_{j,0,0} & g_{j,0,1} & \cdots & g_{j,0,l} & \cdots \\ & & \vdots & & \\ g_{j,k,0} & g_{j,k,1} & \cdots & g_{j,k,l} & \cdots \\ & & \vdots & & \end{pmatrix}.$$

Na sličan način definišu se operatori  $\tilde{H}_j$  i  $\tilde{G}_j$ .

Sada se može dati matični zapis jednačina (2.13)

$$\begin{pmatrix} \tilde{H}_j \\ \tilde{G}_j \end{pmatrix} (H_j^* G_j^*) = \begin{pmatrix} I & 0 \\ 0 & I \end{pmatrix} \quad i \quad (H_j^* G_j^*) \begin{pmatrix} \tilde{H}_j \\ \tilde{G}_j \end{pmatrix} = I. \quad (2.14)$$

**Definicija 4.** Filtri  $H_j, \tilde{H}_j, G_j, \tilde{G}_j$  predstavljaju biortogonalne filtre ako je zadovoljena jednačina (2.14).

Koeficijenti  $c_{j,k}$ , odnosno  $d_{j,k}$  najčešće se nazivaju glatke, odnosno detaljne komponente signala (funkcije) na nivou  $j$ .

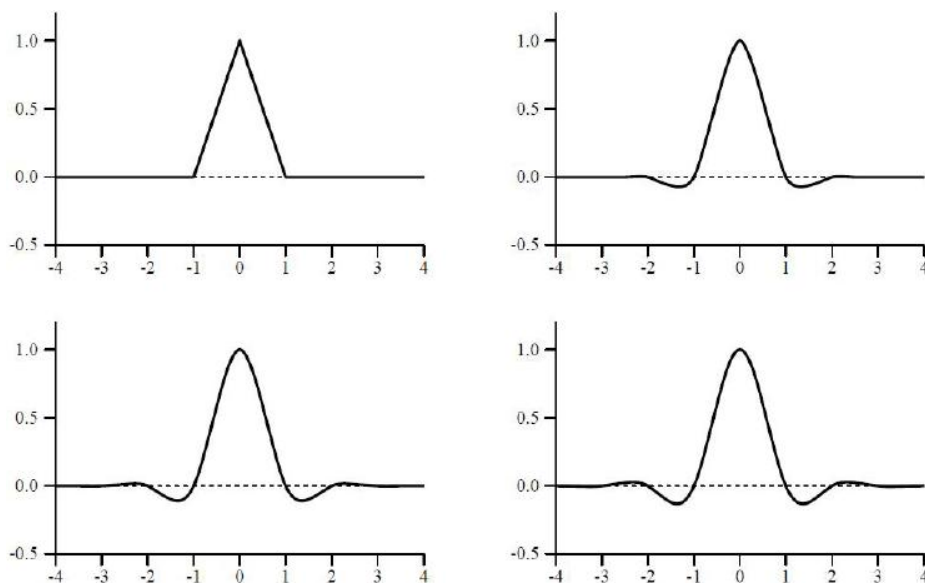
## 2.3 Interpolacioni talasići

Interpolacioni talasići predstavljaju ekstreman slučaj biortogonalnih talasića. Kod njih su dualna funkcija skaliranja i dualni talasić predstavljeni kombinacijom Dirakovih funkcija. Dobra osobina interpolacionih talasića je da su koeficijenti u reprezentaciji funkcijom skaliranja vrednosti fizičke veličine u diadskim tačkama. Stoga se može reći da su fizička interpretacija i reprezentacija funkcijom skaliranja identične.

Interpolacioni talasići su konstruisani na skupu diadskih tačaka na realnoj pravoj:

$$M_j = \{x_{j,k} \in R | x_{j,k} = 2^{-j}k, k \in Z\}, \quad j \in Z,$$

gde su  $x_{j,k}$  mrežne tačke, a  $j$  nivo rezolucije. Primetimo da je  $x_{j-1,k} = x_{j,2k}$ , odakle možemo zaključiti da je  $M_{j-1} \subset M_j$ . Uvedimo sada interpolacione talasiće. Neka su poznate vrednosti funkcije  $f$  na tačkama mreže  $M_j$ , nadjimo vrednosti funkcije u tačkama mreže  $M_{j+1}$  koje ne pripadaju  $M_j$ . Ova procedura ne modifikuje postojeće podatke, tako da se može ponavljati dokle god se ne interpolira funkcija na svim tačkama do željenog nivoa. Interpolaciju postizemo konstrukcijom lokalnih Lagranžovih interpolacionih polinoma  $P_{2N-1}(x)$  reda  $2N-1$  na osnovu  $2N$  najbližih tačaka. Na primer, da bi se odredila vrednost interpolanta u tački  $x_{j+1,2k+1}$ , konstruišemo polinom



Slika 2.1: Interpolacione funkcije skaliranja za  $N=2,4,6,8$

reda  $2N-1$  na osnovu vrednosti funkcije u tačkama  $x_{j,k+l}$  ( $l=-N+1, \dots, N$ ) i računamo vrednost tog polinoma u tački  $x_{j+1,2k+1}$ ,

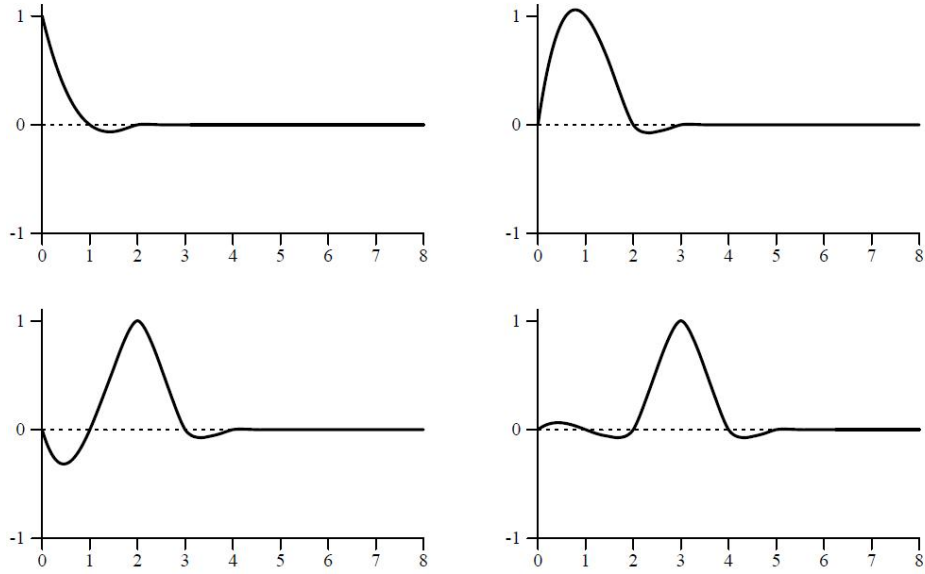
$$f^j(x_{j+1,2k+1}) = \sum_{l=-N+1}^N w_{j,k,l} f(x_{j,k+l}), \quad (2.15)$$

$w_{j,k,l}$  su Lagranževi interpolacioni koeficijenti određeni interpolacijom neparnih tačaka  $x_{j+1,2k+1}$ , uz pomoć parnih  $x_{j+1,2(k+l)} = x_{j,k+l}$ . U slučaju da imamo ograničen interval može se desiti da ovih  $2N$  tačaka ne budu simetrično raspoređene, tada uzimamo  $2N$  najbližih tačaka.

Definišimo sada interpolacionu funkciju skaliranja.

**Definicija 5.** Skup funkcija skaliranja  $\{\varphi_{j,k} | j, k \in \mathbb{Z}\}$  nazivamo interpolacionim ako za mrežne tačke  $x_k$  važi

$$\varphi_{j,k}(x_{k'}) = \delta_{k,k'}, \quad \forall k, k' \in \mathbb{Z},$$



Slika 2.2: Interpolacione funkcije skaliranja za  $N=4$  kada na njih utiče granica

gde  $\delta$  predstavlja Kronekerov simbol

$$\delta_{i,j} = \begin{cases} 1, & i = j \\ 0, & i \neq j \end{cases}.$$

**Teorema 3.** *Ukoliko je skup funkcija skaliranja druge generacije interpolacioni onda važi*

$$\forall k, k' \in Z : h_{j,k,k'} = \delta_{k,k'}.$$

*Dokaz:*

$$\delta_{k,k'} = \varphi_{j,k}(x_{k'}) = \sum_l h_{j,k,l} \varphi_{j+1,l}(x_{k'}) = \sum_l h_{j,k,l} \delta_{l,k'} = h_{j,k,k'}.$$

□

Ovako definisane funkcije omogućavaju da interpolant bude predstavljen u obliku:

$$f^j(x) = \sum_k c_{j,k} \varphi_{j,k}(x), \quad (2.16)$$

gde je  $c_{j,k} = f(x_{j,k})$ . Ukoliko se govori o talasićima prve generacije na ravnomernoj mreži  $M_j$ , sve funkcije skaliranja su translacije i dilatacije funkcije  $\varphi_{0,0}$ . Interpolaciona funkcija skaliranja ima kompaktn nosač, odnosno ona je nula van intervala  $[-2N + 1; 2N - 1]$ .

U svetlu multirezolucije, interpolant  $f^j(x)$  definisan jednačinom (2.16) pripada prostoru  $V_j$ . Ponavljajući postupak za  $j+1$  nivo rezolucije, konstruišemo  $f^{j+1}(x) \in V^{j+1}$ . Iz konstrukcije interpolanta  $f^j(x)$  sledi da je  $f^j(x_{j,k}) = f(x_{j,k})$ . Ranije je navedeno da je  $x_{j,k} = x_{j+1,2k}$ , pa odatle sledi da je  $f^j(x_{j,k}) = f^j(x_{j+1,2k}) = f^{j+1}(x_{j+1,2k})$ . Medjutim, primetimo da je  $f^j(x_{j+1,2k+1}) \neq f^{j+1}(x_{j+1,2k+1})$ . Nazovimo polovinu ove razlike koeficijentima talasića

$$d_{j,k} = 1/2(f^{j+1}(x_{j+1,2k+1}) - f^j(x_{j+1,2k+1}))$$

i postavimo  $\psi_{j,k}(x) = 2\varphi_{j+1,2k+1}(x)$ . Sada može da se definiše funkcija detalja  $d^j(x)$  na sledeći način:

$$d^j(x) = \sum_m d_{j,m} \psi_{j,m}(x).$$

Odavde se vidi da je  $f^{j+1}(x) = f^j(x) + d^j(x)$ , odnosno funkcija  $d^j(x)$  predstavlja razliku izmedju  $f^{j+1}(x)$  i  $f^j(x)$ . Koristeći ovu jednačinu i jednačinu (2.16) dobija se:

$$\sum_k c_{j+1,k} \varphi_{j+1,k}(x) = f^j(x) + \sum_m d_{j,m} \psi_{j,m}(x).$$

Stavljajući da je  $x = x_{j+1,2k+1}$  i aproksimirajući  $f^j(x)$  jednačinom (2.15), dobija se:

$$c_{j+1,2k+1} = \sum_{l=-N-1}^N w_{j,k,l} f(x_{j,k+l}) + 2d_{j,k}.$$

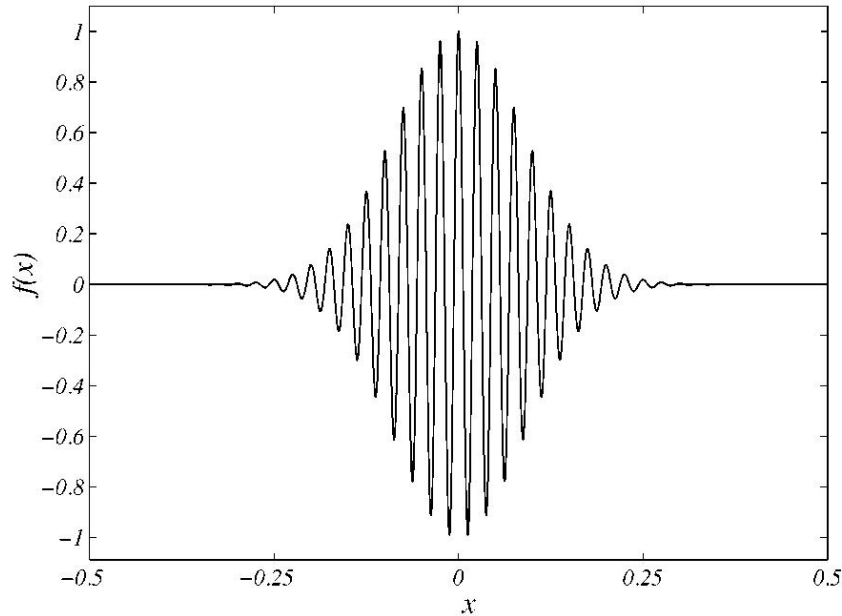
Kako je  $x_{j,k+l} = x_{j+1,2k+2l}$ , a iz jednačine (2.16) imamo da je  $c_{j,k} = f(x_{j,k})$ , dobijamo brzu transformaciju talasićima.

$$d_{j,k} = \frac{1}{2} \left( c_{j+1,2k+1} - \sum_l w_{j,k,l} c_{j+1,2k+2l} \right),$$

$$c_{j,k} = c_{j+1,2k}.$$

Inverzna transformacija je data sa:

$$c_{j+1,2k} = c_{j,k} \quad c_{j+1,2k+1} = 2d_{j,k} + \sum_l w_{j,k,l} c_{j,k+l},$$



Slika 2.3:  $f(x) = \cos(80\pi x) \exp^{-64x^2}$

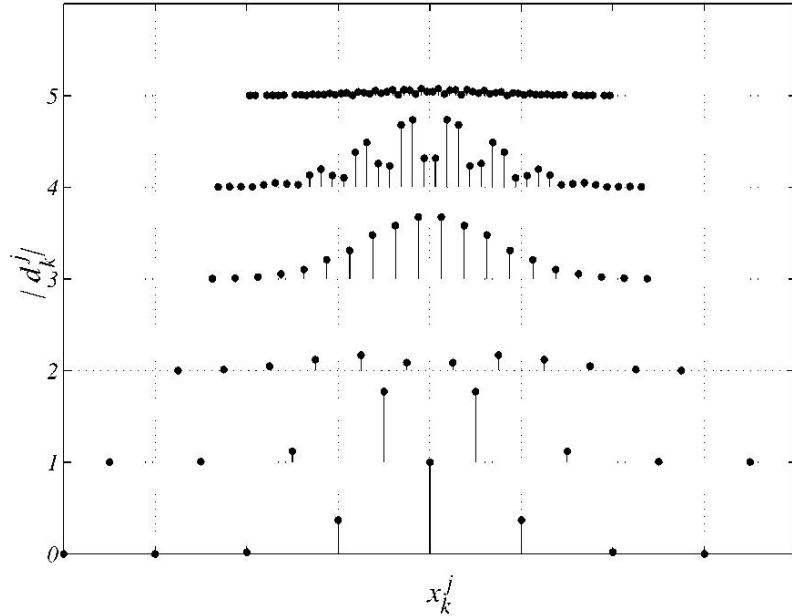
gde su  $w_{j,k,l}$  interpolacioni koeficijenti pri interpoliranju neparnih tačaka  $x_{j+1,2k+1}$  uz pomoć parnih  $x_{j+1,2(k+l)} = x_{j,k+l}$  uvedeni relacijom (2.15).

Ovo je postupak za konstrukciju interpolacionih talasića na realnoj pravnoj, i primer prve generacije talasića. Postupak je isti i ukoliko je interval ograničen ili mreža neravnomerna, s tim što se tada koriste talasići druge generacije.

Medjutim, interpolacioni talasići imaju i velike nedostatke. Bazis talasića konstruisan uz pomoć interpolacione funkcije skaliranja nije Risov bazis za  $L_2$ , isto tako talasići nemaju momente jednake nuli. Problem predstavlja i to što su dualni talasići Dirakove funkcije koje ne pripadaju  $L_2$ .

Razmotrimo sad primer koji će pokazati još jednu manu interpolacionih talasića. Posmatrajmo funkciju  $f(x) = \cos(80\pi x) \exp^{-64x^2}$  (slika 2.3). Koeficijenti transformacije interpolacionim talasićima ove funkcije dati su na sledećoj slici 2.4. Primetno je prisustvo velikih koeficijenata talasića na nižim nivoima transformacije.

Neki od ovih problema biće rešeni u sledećem poglavlju konstrukcijom talasića druge generacije pomoću lifting šeme.



Slika 2.4: Distribucija koeficijenata  $d_{j,k}$  (prikazani samo koeficijenti za koje je  $|d_{j,k}| > 10^{-3}$ )

## 2.4 Lifting šema

Lifting šema predstavlja način za konstrukciju talasića druge generacije. Osnovna ideja liftinga je da se krene od proste multirezolucije i postepeno gradi multirezolucija sa unapred zadatim svojstvima. Lifting šemu možemo videti i kao modifikaciju talasića dobijenu oduzimanjem od postojećeg talasića ( $\psi^{old}$ ) linearne kombinacije funkcija skaliranja na istom nivou rezolucije

$$\psi(x) = \psi^{old}(x) - \sum_k u_k \varphi(x - k).$$

Popravka  $u_k$  bira se tako da novi talasić ima željena svojstva. Ovo će ostaviti funkciju skaliranja nepromenjenu, dok će talasić biti promenjen. Lifting bi tako trebalo da omogući da imamo transformaciju talasićima bolje performanse. Konkretno, Sveldens je pokazao [13, 14] da interpolacioni talasić reda  $N$  može biti modifikovan lifting šemom tako da rezultujući talasić ima  $N$  momenata jednakih nuli.

Koeficijenti (filtri)  $h, g, \tilde{h}, \tilde{g}$  liftingom su modifikovani na sledeći način:

$$\begin{aligned} h_{j,k,l} &= h_{j,k,l}^{old}, \\ \tilde{h}_{j,k,l} &= \tilde{h}_{j,k,l}^{old} + \sum_m u_{j,k,m} \tilde{g}_{j,m,l}^{old}, \\ g_{j,k,l} &= g_{j,k,l}^{old} - \sum_m u_{j,k,m} h_{j,m,l}^{old}, \\ \tilde{g}_{j,k,l} &= \tilde{g}_{j,k,l}^{old}. \end{aligned}$$

Dokažimo sada da su i novi filtri  $h, \tilde{h}, g, \tilde{g}$  biortogonalni. Uz to ćemo dati matični oblik lifting šeme.

**Teorema 4.** *Neka je dat skup biortogonalnih filtara  $H_j^{old}, \tilde{H}_j^{old}, G_j^{old}, \tilde{G}_j^{old}$ . Tada su filtri  $H_j, \tilde{H}_j, G_j, \tilde{G}_j$  dati sa:*

$$\begin{aligned} H_j &= H_j^{old}, \\ G_j &= G_j^{old} - U_j^* H_j^{old}, \\ \tilde{H}_j &= \tilde{H}_j^{old} + U_j \tilde{G}_j^{old}, \\ \tilde{G}_j &= \tilde{G}_j^{old}, \end{aligned}$$

biortogonalni. Matrica  $U_j$  predstavlja matični zapis,  $U_j = \{u_{j,k,m}\}$ .

*Dokaz:* Napišimo lifting šemu u matičnom obliku

$$\begin{aligned} \begin{pmatrix} \tilde{H}_j \\ \tilde{G}_j \end{pmatrix} &= \begin{pmatrix} 1 & U_j \\ 0 & 1 \end{pmatrix} \begin{pmatrix} \tilde{H}_j^{old} \\ \tilde{G}_j^{old} \end{pmatrix}, \\ \begin{pmatrix} H_j \\ G_j \end{pmatrix} &= \begin{pmatrix} 1 & 0 \\ -U_j^* & 1 \end{pmatrix} \begin{pmatrix} H_j^{old} \\ G_j^{old} \end{pmatrix}. \end{aligned}$$

Konjugovanjem se dobija:

$$\begin{pmatrix} H_j^* & G_j^* \end{pmatrix} = \begin{pmatrix} H_j^{*old} & G_j^{*old} \end{pmatrix} \begin{pmatrix} 1 & -U_j \\ 0 & 1 \end{pmatrix}.$$

Odavde, prvi uslov iz (2.14) možemo zapisati:

$$\begin{aligned} \begin{pmatrix} 1 & U_j \\ 0 & 1 \end{pmatrix} \begin{pmatrix} \tilde{H}_j^{old} \\ \tilde{G}_j^{old} \end{pmatrix} \begin{pmatrix} H_j^{*old} & G_j^{*old} \end{pmatrix} \begin{pmatrix} 1 & -U_j \\ 0 & 1 \end{pmatrix} &= \\ = \begin{pmatrix} 1 & U_j \\ 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & -U_j \\ 0 & 1 \end{pmatrix} &= \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \end{aligned}$$

a drugi uslov iz (2.14):

$$(H_j^{*old} \quad G_j^{*old}) \begin{pmatrix} 1 & -U_j \\ 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & U_j \\ 0 & 1 \end{pmatrix} \begin{pmatrix} \tilde{H}_j^{old} \\ \tilde{G}_j^{old} \end{pmatrix} = 1.$$

□

Promenom u jednačinama (2.10) i (2.11), u skladu sa liftingom, dobija se brza transformacija liftovanim talasićima:

$$d_{j,m} = \sum_l \tilde{g}_{j,m,l}^{old} c_{j+1,l},$$

$$c_{j,m} = \sum_l \tilde{h}_{j,k,l}^{old} c_{j+1,l} + \sum_m u_{j,k,m} d_{j,m}.$$

Inverzna transformacija dobija se iz jednačine (2.12):

$$c_{j+1,k} = \sum_l h_{j,k,l}^{old} \left( c_{j,l} - \sum_m u_{j,k,m} d_{j,m} \right) + \sum_k g_{j,k,l}^{old} d_{j,k}.$$

Razmotrimo sada šta se dešava kada lifting šemu primenimo na interpolacione talasiće. Transformacija interpolacionim talasićima se može značajno unaprediti ukoliko se primeni dodatni lifting korak. Konkretno, neki momenti postaju jednaki nuli, ali i dualne funkcije skaliranja i dualni talasići pripadaju  $L_2$ . Postavimo da je  $u_{j,k,m} = \tilde{w}_{j,k,m-k}$ , gde su  $\tilde{w}_{j,k,l}$  (slično kao i  $w_{j,k,l}$ ) interpolacioni koeficijenti pri interpoliranju parnih tačaka  $x_{j+1,2k}$  uz pomoć neparnih  $x_{j+1,2k+2l+1}$ .

Primenom liftinga, brza transformacija liftovanim talasićima je oblika:

$$d_{j,k} = \frac{1}{2} \left( c_{j+1,2k+1} - \sum_l w_{j,k,l} c_{j+1,2k+2l} \right), \quad (2.17)$$

$$c_{j,k} = c_{j+1,2k} + \sum_l \tilde{w}_{j,k,l} d_{j,k+l}, \quad (2.18)$$

a inverzna transformacija je data sa

$$c_{j+1,2k} = c_{j,k} - \sum_l \tilde{w}_{j,k,l} d_{j,k+l}, \quad (2.19)$$

$$c_{j+1,2k+1} = 2d_{j,k} + \sum_l w_{j,k,l} c_{j+1,2k+2l}, \quad (2.20)$$

gde su  $w_{j,k,l}$  i  $\tilde{w}_{j,k,l}$  ranije definisani.

Primetimo da red interpolacionog polinoma koji koristi parne i onog koji koristi neparne tačke u interpolaciji ne moraju da budu isti. Drugim rečima koeficijenti  $\tilde{w}_{j,k,l}$  mogu biti konstruisani iz interpolacionog polinoma reda  $(2\tilde{N}-1)$  koji koristi  $2\tilde{N}$  susednih neparnih tačaka. Rezultat ovoga je da se transformacija liftovanim interpolacionim talasićima kontroliše sa dva parametra  $N$  i  $\tilde{N}$ . Sveldens je pokazao [13],[14] da  $N$  kontroliše broj nula momenata interpolacione funkcije skaliranja, dok  $\tilde{N}$  kontroliše broj nula momenata interpolacionih talasića. Može se pokazati:

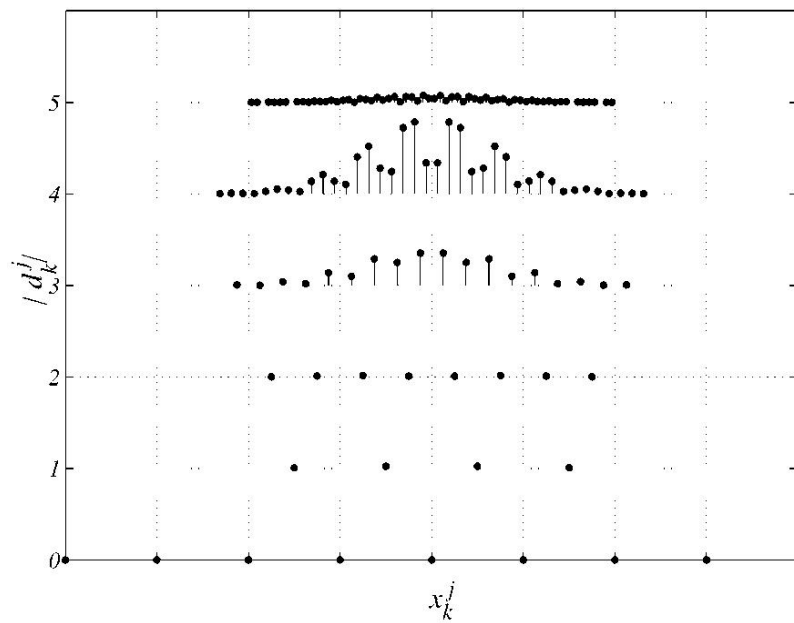
$$\int_D x^p \varphi(x) dx = \delta_{p,0} \quad za \quad 0 \leq p \leq 2N - 1,$$

$$\int_D x^p \psi(x) dx = 0 \quad za \quad 0 \leq p \leq 2\tilde{N} - 1,$$

pri čemu je  $\int_D$  integral nad domenom za koji su talasići konstruisani. Najbolja kompresija se postiže za  $\tilde{N} = N$  [17]. Primetimo da ukoliko je  $\tilde{N} = 0$  dobija se standardna transformacija interpolacionim talasićima. Momenti jednaki nuli određuju do kog stepena je moguće tačno zapisati polinome bazisnim funkcijama. Ukoliko je moment talasića jednak nuli do reda  $k$ , za bilo koji polinom stepena manjeg od  $k$  svi koeficijenti talasića su jednaki nuli. To znači da funkcija pripada prostoru funkcije skaliranja  $V$ , odnosno da funkcija skaliranja ima svojstvo reprezentacije polinoma do stepena  $k-1$ . Više o momentima jednakim nuli u [9].

Da bismo uvideli prednosti liftovanih interpolacionih talasića, primenimo transformaciju na funkciju  $f(x) = \cos(80\pi x) \exp^{-64x^2}$  primeru u poglavlju interpolacioni talasići.

Primenom ove transformacije, na nižim nivoima transformacije nemamo velike koeficijente (slika 2.5).



Slika 2.5: Distribucija koeficijenata  $d_{j,k}$  (prikazani samo koeficijenti za koje je  $|d_{j,k}| > 10^{-3}$ )

# 3

## Primena na diferencijalne jednačine

### 3.1 Numeričko rešavanje diferencijalnih jednačina

U ovom poglavlju su izložena dva algoritma za rešavanje diferencijalnih jednačina (DJ). Diferencijalne jednačine rešavamo aproksimiranjem izvoda u tačkama kolokacije. Kako bi algoritam bio što efikasniji, nećemo raditi sa svim tačkama. Umesto toga, korišćićemo dinamičku adaptaciju mreže. Već smo definisali mrežu kao skup tačaka:

$$M_j = \{x_{j,k} \in R \mid x_{j,k} = 2^{-j}k, k \in Z\}, \quad j \in Z,$$

pri čemu umesto  $R$  može biti zadat bilo koji zatvoreni interval  $\Omega$ . Za mrežne tačke važi da je  $x_{j,k} = x_{j+1,2k}$ , što garantuje da je  $M_j \subset M_{j+1}$ . Na osnovu ranijih razmatranja funkciju  $u$  na  $J$ -tom nivou možemo predstaviti izrazom:

$$u^J(x) = \sum_{k \in Z} c_{0,k} \varphi_{0,k}(x) + \sum_{j=0}^{J-1} \sum_{l \in Z} d_{j,l} \psi_{j,l}(x).$$

Sada dolazi do izražaja prednost talasića. Ukoliko se u aproksimaciji izostave talasići čiji koeficijenti su manji od nekog zadatog  $\epsilon$ , zadržaće se dobra aproksimacija  $u_{\geq}^J(x)$ , gde je

$$u^J(x) = u_{\geq}^J(x) + u_{<}^J(x),$$
$$u_{\geq}^J(x) = \sum_{k \in Z} c_{0,k} \varphi_{0,k}(x) + \sum_{j=0}^{J-1} \sum_{|d_{j,l}| \geq \epsilon} d_{j,l} \psi_{j,l}(x), \quad (3.1)$$

$$u_{<}^J(x) = \sum_{j=0}^{J-1} \sum_{|d_{j,l}| < \epsilon} d_{j,l} \psi_{j,l}(x).$$

Greška ove aproksimacije je ocenjena u sledećoj teoremi.

**Teorema 5.** *Za svako  $\epsilon > 0$  postoji pozitivna konstanta  $C_1$  takva da važi:*

$$\|u^J(x) - u_{\geq}^J(x)\|_{L^2} = \|u_{<}^J(x)\|_{L^2} \leq \epsilon C_1.$$

*Dokaz:*

$$\begin{aligned} \|u_{<}^J(x)\|_{L^2} &= \left\| \sum_{j=0}^{J-1} \sum_{|d_{j,l}| < \epsilon} d_{j,l} \psi_{j,l}(x) \right\|_{L^2} \leq \\ &\leq \sum_{j=0}^{J-1} \sum_{|d_{j,l}| < \epsilon} |d_{j,l}| \|\psi_{j,l}(x)\|_{L^2} \leq \epsilon N_W \|\psi(x)\|_{L^2} \end{aligned}$$

pri čemu je  $N_W$  ukupan broja talasića. Važi  $|d_{j,l}| < \epsilon$ , odatle  $\epsilon$  u poslednjoj nejednakosti.  $\square$

Broj značajnih koeficijenata talasića  $N_s$  je prema [2] ograničen sledećom nejednakošću:

$$N_s \leq C_2 \epsilon^{-1/2N}.$$

Kombinovanjem prethodne dve nejednakosti dobijamo ocenu:

$$\|u^J(x) - u_{\geq}^J(x)\|_{L^2} \leq C_3 N_s^{-2N}. \quad (3.2)$$

Ukoliko je nivo multirezolucije  $J$  dovoljno velik i greška  $\|u^J(x) - u_{\geq}^J(x)\|_{L^2}$  je zanemarljiva, odnosno, za dovoljno veliko  $J$  red greške je isti kao i red greške računara pri izračunavanju. Ovo sledi indirektno iz jednakosti (3.2), jer broj značajnih koeficijenata talasića direktno zavisi od nivoa multirezolucije  $J$ .

Shodno tome, funkcija  $u$  može se predstaviti uz pomoć manje koeficijenata, zadržavajući dobru aproksimaciju. Prednost kompresije talasićima je što se pomoću  $N_s$  mrežnih tačaka može rekonstruisati funkcija  $u_{\geq}^J$ . Primećimo da je svaka funkcija skaliranja  $\varphi_{j,k}(x)$  jedinstveno povezana sa tačkom  $x_{j,k}$ , a svaki talasić  $\psi_{j,l}$  je jedinstveno povezan sa tačkom  $x_{j+1,2l+1}$ . Kada je izvršena dekompozicija talasićima, svaka mrežna tačka na najfinijem nivou rezolucije  $J$  je jedinstveno povezana, ili sa nekim od talasića, ili sa funkcijom skaliranja na najgrubljem nivou rezolucije. U skladu sa tim, tačka kolokacije će biti izostavljena iz mreže, ukoliko je odgovarajući talasić izostavljen iz

aproksimacije. Ovde treba napomenuti da, zbog stabilnosti algoritma rekonstrukcije, treba zadržati sve mrežne tačke povezane sa funkcijom skaliranja na najgrubljem nivou.

Treba voditi računa i o tačkama koje mogu postati značajne. U adaptivnu mrežu, treba uključiti i tačke koje su povezane sa talasićima, koji pripadaju susednim oblastima talasića. Za talasić  $\psi_{j',l}(x)$  kazemo da pripada susednoj oblasti talasića  $\psi_{j,k}(x)$  ukoliko je:

$$|j - j'| \leq L \quad \left| 2^{j'-j}k - l \right| \leq F, \quad (3.3)$$

pri čemu  $L$  definiše koliko finijih i grubljih nivoa bi trebalo uključiti, dok  $F$  definiše širinu susedne oblasti. Vrednosti  $L$  i  $F$  utiču na konačan broj tačaka kolokacije prisutnih u  $M_{\geq}$ . Radi efikasnosti poželjno je da taj broj bude što manji. Vasiljev [17] predlaže da se uzme  $L=F=1$ , odnosno da se za susedne tačke uzimaju samo najbliže tačke na istom nivou i na finijem nivou multirezolucije. Nema potrebe za uključivanjem tačaka sa grubljeg nivoa.

Sada ćemo predstaviti algoritme. Prvi algoritam kreće od najfinijeg nivoa i zatim izbacuje nepotrebne tačke. Drugi algoritam kreće od najgrubljeg nivoa i zatim dodaje potrebne tačke.

Algoritam za prvi pristup se sastoji od sledećih koraka:

1. Uneti željeni nivo multirezolucije, kao i korak na najgrubljem nivou (korak na najfinijem nivou = korak na najgrubljem \*  $2^{-J}$ ).
2. Rešiti problem na najfinijem nivou rezolucije i staviti  $c_{J,k} = u(x_{J,k})$ . Kako bi poboljšali brzinu programa izvode aproksimirati konačnim razlikama prvog reda.
3. Izvršiti transformaciju liftovanim talasićima, nakon čega su dobijene sve vrednosti  $c_{0,k}$  i  $d_{j,l}$ ,  $0 \leq j \leq J - 1$ .
4. Analizirati koeficijente  $d_{j,l}$ , kreirati masku  $O$  sa mrežnim tačkama  $x_{j,k}$ , koje su odgovaraju talasićima za koje važi  $|d_{j,l}| \geq \epsilon$ .
5. Uključiti u masku  $O$  i sve tačke koje su povezane sa funkcijama skaliranja na najgrubljem nivou.
6. Proširiti masku  $O$  sa tačkama koje su povezane sa susednim talasićima.
7. Dobijena je adaptirana mreža i tačke koje su ostale predstavljaju tačke kolokacije. Rešiti problem podeljenim razlikama koristeći  $2N + 1$  tačaka.

Drugi pristup zasniva se na ideji da se u svakoj iteraciji popravljaju mreža, tj. da se krene od najgrubljeg nivoa i da sam program definiše sledeće nivoe na osnovu veličine koeficijenata talasića. Algoritam se sastoji od sledećih koraka:

1. Uneti korak na najgrubljem nivou ( $V_0$ ), automatski napraviti sledeći multirezolucijski nivo ( $V_1$ ). Potrebno je imati bar dva nivoa kako bi se našli koeficijenti talasića.
2. Rešiti problem na najvišem multirezolucijskom nivou koji postoji u tom trenutku korišćenjem podeljenih razlika (u sledećem poglavlju biće objašnjene podeljene razlike na neravnomernoj mreži). Sada postaviti  $c_{J,k} = u(x_{J,k})$ .
3. Izvršiti transformaciju interpolacionim talasićima, nakon čega su dobijene sve vrednosti  $c_{0,k}$  i  $d_{j,l}$ ,  $0 \leq j \leq J - 1$ .
4. Analizirati koeficijente  $d_{j,l}$ , kreirati masku O sa mrežnim tačkama  $x_{j,k}$ , koje su povezane sa talasićima za koje važi  $|d_{j,l}| \geq \epsilon$ .
5. Uključiti u masku O i sve tačke koje su povezane sa funkcijama skaliranja na najgrubljem nivou.
6. Proširiti masku O sa tačkama koje su povezane sa susednim talasićima.
7. Dodati još jedan multirezolucijski nivo, na kojem će tačke biti gušće nego na najfinijem nivou do tada.
8. Ukoliko nema razlike između nivoa  $V_J$  i  $V_{J-1}$  stati, u suprotnom, ponovo se vratiti na korak 2.

Objasnimo detaljnije korake 6 i 7. Posmatrajmo talasić  $\psi_{j,k}$ , on je povezan sa tačkom  $x_{j+1,2k+1}$ . Njemu susedni talasići na osnovu (23) su  $\psi_{j,k\pm 1}$  na istom multirezolucijskom nivou, a na višem  $\psi_{j+1,2k}$  i  $\psi_{j+1,2k\pm 1}$ . Shodno tome, u mrežu treba uključiti tačke  $x_{j+1,2k+1}$ ,  $x_{j+1,2k-1}$ ,  $x_{j+1,2k+3}$ ,  $x_{j+2,4k-1}$ ,  $x_{j+2,4k+1}$ ,  $x_{j+2,4k+3}$ . Ovo predstavlja korak 6. za  $j \leq J - 2$ . Kada je  $j = J - 1$ , ovo je korak 7, uključujemo po istom principu tačke  $x_{j+1,2k+1}$ ,  $x_{j+1,2k-1}$ ,  $x_{j+1,2k+3}$ , a tačke  $x_{j+2,4k-1}$ ,  $x_{j+2,4k+1}$ ,  $x_{j+2,4k+3}$  daju novi multirezolucijski nivo J+1 zajedno sa svim tačkama sa J nivoa.

Objasnimo sada i korak 8. Profinjavanjem mreže, koeficijenti talasića na višim nivoima postaju sve manji. Kada na nekom nivou više ne bude bilo  $d_{j,k} \geq \epsilon$  korak 7 neće dodavati nove tačke, samim tim neće biti razlike između prostora  $V_j$  i  $V_{j+1}$ .

Nedostatak drugog algoritma u odnosu na prvi, je što se u svakom prolasku kroz algoritam traži rešenje sistema linearnih jednačina primenom singularne dekompozicije. Sa druge strane, velika prednost je što se radi sa manjim brojem tačaka. Primera radi, neka je broj tačaka na najgrubljem nivou ( $j=0$ ) 11 i broj nivoa  $J=5$ . Tada je broj tačaka na najfinijem nivou ( $j=5$ ) multirezolucije  $321$  ( $(11 - 1) * 2^5 + 1$ ). Sa prvim algoritmom treba rešiti sistem linearnih jednačina dimenzije  $321 \times 321$ , aproksimirajući izvode konačnim razlikama prvog reda, potom rešiti sistem  $N_{s_J} \times N_{s_J}$  podeljenim razlikama korišćenjem  $2N+1$  tačaka ( $N_{s_j}$  broj tačaka koji je ostao na najfinijem nivou). U drugom algoritmu rešavamo u svakom koraku sistem  $N_{s_J} \times N_{s_J}$ .

Takodje u drugom algoritmu ne zadaje se  $J$ , već algoritam uzima  $J=2$  u prvoj iteraciji i zatim se u svakom prolasku kroz algoritam povećava. Iz navedenog sledi da je bolje koristiti drugi algoritam kada se izbacuje dosta tačaka.

### 3.2 Aproksimacija izvoda

Prilikom rešavanja DJ metodom kolokacije potrebno je dobiti izvode funkcije u tačkama kolokacije. Za aproksimiranje izvoda koristimo se podeljenim razlikama. Posmatrajmo Lagranžev interpolacioni polinom

$$p(x) = \sum_{j=i-N}^{i+N} l_j(x)u(x_j),$$

pri čemu je

$$l_j(x) = \frac{P_{i,j}(x)}{P_{i,j}(x_j)}, \quad P_{i,j}(x) = \prod_{l=i-N, l \neq j}^{i+N} (x - x_l).$$

Tačke  $x_i$ , ukoliko se radi o ravnomernoj mreži, su oblika:

$$x_N = x, \quad x_{N-1} = x - h, \quad x_{N+1} = x + h, \quad \dots$$

$$x_0 = x - Nh, \quad x_{2N} = x + Nh.$$

Diferenciranjem Lagranževog interpolacionog polinoma dobijamo da je:

$$p'(x) = \sum_{j=i-N}^{i+N} l'_j(x)u(x_j).$$

Sada aproksimacija podeljenim razlikama funkcije  $u'(x)$  u središnjoj tački  $x = x_N$  ima oblik:

$$u'(x_N) \approx l'_0(x_N)u(x_0) + \dots +$$

$$+ l'_{N-1}(x_N)u(x_{N-1}) + l'_N(x_N)u(x_N) + l'_{N+1}(x_N)u(x_{N+1}) + \dots +$$

$$+ l'_{2N}(x_N)u(x_{2N}) + O(h^{2N-1}).$$

Analogno, izvod reda  $d$  aproksimiramo izrazom:

$$u^d(x) \approx \sum_{j=i-N}^{i+N} u(x_j) \frac{P_{i,j}^d(x)}{P_{i,j}^d(x_j)}.$$

Pri čemu su:

$$P'_{i,j}(x) = \sum_{l=i-N, l \neq j}^{i+N} \prod_{m=i-N, m \neq j, l}^{i+N} (x - x_m),$$

$$P''_{i,j}(x) = \sum_{l=i-N, l \neq j}^{i+N} \sum_{m=i-N, m \neq j, l}^{i+N} \prod_{n=i-N, n \neq j, l, m}^{i+N} (x - x_n).$$

Primetimo da ukoliko se radi o ravnomernoj mreži, dobija se klasična aproksimacija konačnim razlikama. Kada aproksimiramo izvode dobijamo sistem linearnih jednačina dimenzije  $N_J \times N_J$  ( $N_J$  predstavlja broj tačaka na najfinijem multirezolucijskom nivou). Sistem rešavamo singularnom dekompozicijom zbog najčešće loše uslovljenosti matrice sistema.

Opišimo sada proceduru za nalaženje izvoda. U literaturi se najčešće koriste sledeći pristupi.

Diferenciranje jednačine (2.16) i procenjivanje na mreži  $M_{\geq}$ . Nedostatak ovog pristupa je što se zahteva procena uticaja talasća na svim nivoima, usled čega je efikasnost transformacije talasićima izgubljena.

Drugi pristup je metod podeljenih razlika na neregularnim mrežama. Nedostatak ovog pristupa je što koristi transformaciju talasićima samo za adaptaciju mreže, tako da nisu iskorišćene sve prednosti multirezolucijske dekompozicije. Ovaj pristup zastupa i Holmstrom [4], i biće korišćen u ovom radu.

Objasnimo i pristup koji koristi Vasiljev. Njegova ideja je da se talasići koriste za ono za šta su najbolji, a to je za interpolaciju i kompresiju, a operatori podeljenih razlika za diferenciranje polinoma.

Proces diferenciranja je zasnovan na interpolacionim svojstvima druge generacije interpolacionih talasića. Podsetimo se da koeficijenti talasića  $d_{j,k}$  predstavljaju razliku izmedju aproksimacije funkcije na  $j+1$  nivou i aproksimacije na  $j$  nivou multirezolucije. Ukoliko nema tačaka u okolini  $x_{j,k}$ , tj.  $|d_{j,k+l}| < \epsilon$  ( $l=-1,0$ ), tačke  $x_{j+1,2k\pm 1}$  nisu prisutne u mreži  $M_{j+1, \geq}$ , te postoji neka okolina  $\Omega_{j,k}$  tačke  $x_{j,k}$ , u kojoj je funkcija dobro aproksimirana lokalnim polinomom:

$$\left| f(x) - \sum c_{j,l} \varphi_{j,l}(x) \right| \leq C_4 \epsilon, \quad x \in \Omega_{j,k}.$$

Diferenciranjem ovog polinoma dobija se vrednost izvoda na toj lokaciji. Označimo sa  $D_{j, \geq}$  kolekciju takvih tačaka na  $j$ -tom nivou. Nalaženje izvoda u svim tačkama mreže  $M_j$  se sastoji iz sledećih koraka:

1. Znajući vrednost rešenja u svim tačkama adaptivne mreže  $M_{\geq}$  izvršiti transformaciju talasićima.
2. Analizirati tačke mreže i napraviti kolekciju  $D_{j, \geq}$ .
3. Rekonstruisati funkciju počevši od najgrubljeg nivoa. Na svakom nivou  $j$

naći izvode funkcije u tačkama koje pripadaju  $D_{j,\geq}$ .

Po završetku ovog postupka imaćemo izvode u svim tačkama.

Razmotrimo sada tačnost ovakvog načina diferenciranja. Već smo rekli da je greška operatora konačnih razlika u slučaju ravnomerne mreže  $O(h^{2N-1})$ , kod aproksimacije prvog izvoda konačnim razlikama, dok je greška Lagranževog interpolacionog polinoma  $O(h^{2N})$ . Slično kao i u jednakosti (2.17), zbog smanjivanja reda tačnosti diferenciranjem:

$$|Du^J(x) - Du_{\geq}^J(x)| \leq C_5 N_s^{-2N+1},$$

pri čemu D označava operator diferenciranja.

### 3.3 Numerički rezultati

U ovom poglavlju biće prikazani rezultati testiranja algoritama na primerima. Kao test primeri izabrani su karakteristični singularno-perturbovani problemi. U prvom primeru singularitet se nalazi na oba kraja intervala, u drugom samo na levom kraju, a u trećem primeru je singularitet na sredini intervala.

Prvi primer koji testiramo razmatran je u [8]:

$$-\theta u'' + u = -\cos^2(\pi x) - 2\theta\pi^2 \cos(2\pi x).$$

uz granične uslove:

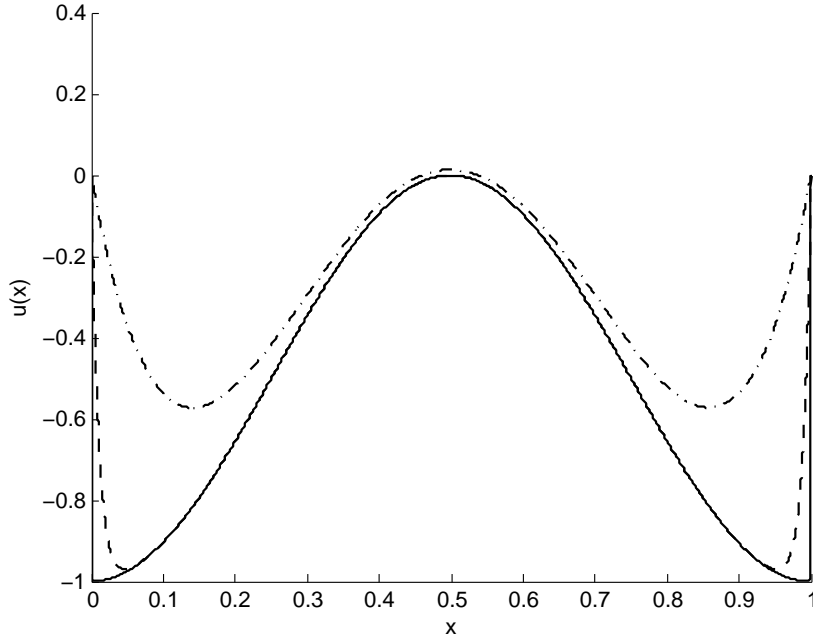
$$u(0) = 0 \quad i \quad u(1) = 0.$$

Analitičko rešenje ovog problema je:

$$u(x) = \frac{e^{-\frac{1-x}{\sqrt{\theta}}} + e^{-\frac{x}{\sqrt{\theta}}}}{1 + e^{-\frac{1}{\sqrt{\theta}}}} - \cos^2(\pi x).$$

Testiran je prvi algoritam na ovom problemu za vrednosti  $\theta=0.01, 0.001, 0.0001, 0.0000001$ . Za stepen interpolacionog polinoma uzeto je  $N = 1, 3$ , a za prag kompresije  $\epsilon = 0.1, 0.001$ . Svi rezultati su dobijeni sa 5 nivoa multirezolucije i početnim korakom jednakim 0.1.

Za  $N=1$  prvi algoritam nema mnogo smisla. Koristimo konačne razlike na najfinijem nivou multirezolucije da bismo dobili transformaciju talasićima. Potom koristeći manje tačaka uz pomoć podeljenih razlika za  $N=1$  tražimo rešenje. Prema tome, rešenje koje već imamo neće se poboljšati.



Slika 3.1: Analitičko rešenje primera 1. za  $\theta = 0.01$ (-),  $\theta = 0.0001$ (--),  $\theta = 0.0000001$ (-)

Medjutim, za  $N > 1$ , prvi algoritam počinje od najfinijeg nivoa i za  $N=1$  nalazi rešenje uz pomoć konačnih razlika, a zatim to rešenje koristi za transformaciju talasićima. Potom, koristeći manje tačaka, nalazimo rešenje uz pomoć podeljenih razlika za  $N$ . Prednost kada je  $N > 1$  je u tome što početno rešenje tražimo za  $N=1$ , te imamo manje izračunavanja. Kada odredimo koje tačke su značajne podeljenim razlikama za  $N > 1$  nalazimo rešenje.

Tabela 1. prikazuje uporednu analizu grešaka, za različite vrednosti parametara  $\theta$  i  $\epsilon$ . Za vrednost koraka na najgrubljem nivou uzeto je 0.1 i 5 nivoa multirezolucije. Kolona  $\|u - u^J\|_\infty$  predstavlja grešku aproksimacije u odnosu na analitičko rešenje, po normi zadatoj na sledeći način

$$\|u(x) - u^J(x)\| = \max_{i \in I} |u(x_i) - u^J(x_i)|.$$

Skup  $I$  predstavlja skup indeksa na najfinijem nivou multirezolucije.

Kolona  $\|u - u^J\|_2$  predstavlja grešku aproksimacije u odnosu na ana-

litičko rešenje po normi zadatoj na sledeći način.

$$\|u - u^J\|_2 = \frac{1}{n} \left( \sum_{k=0}^n (u(x_k) - u^J(x_k))^2 \right)^{\frac{1}{2}}.$$

U svakoj vrsti imamo po dva broja. Prvi broj predstavlja rezultate konačnih razlika za  $N=1$  i ravnomernu mrežu. Drugi broj predstavlja rezultate pode-  
ljenih razlika za  $N=3$  i neravnomernu mrežu.

Tabela 1. Rezultati prvog algoritma na primeru

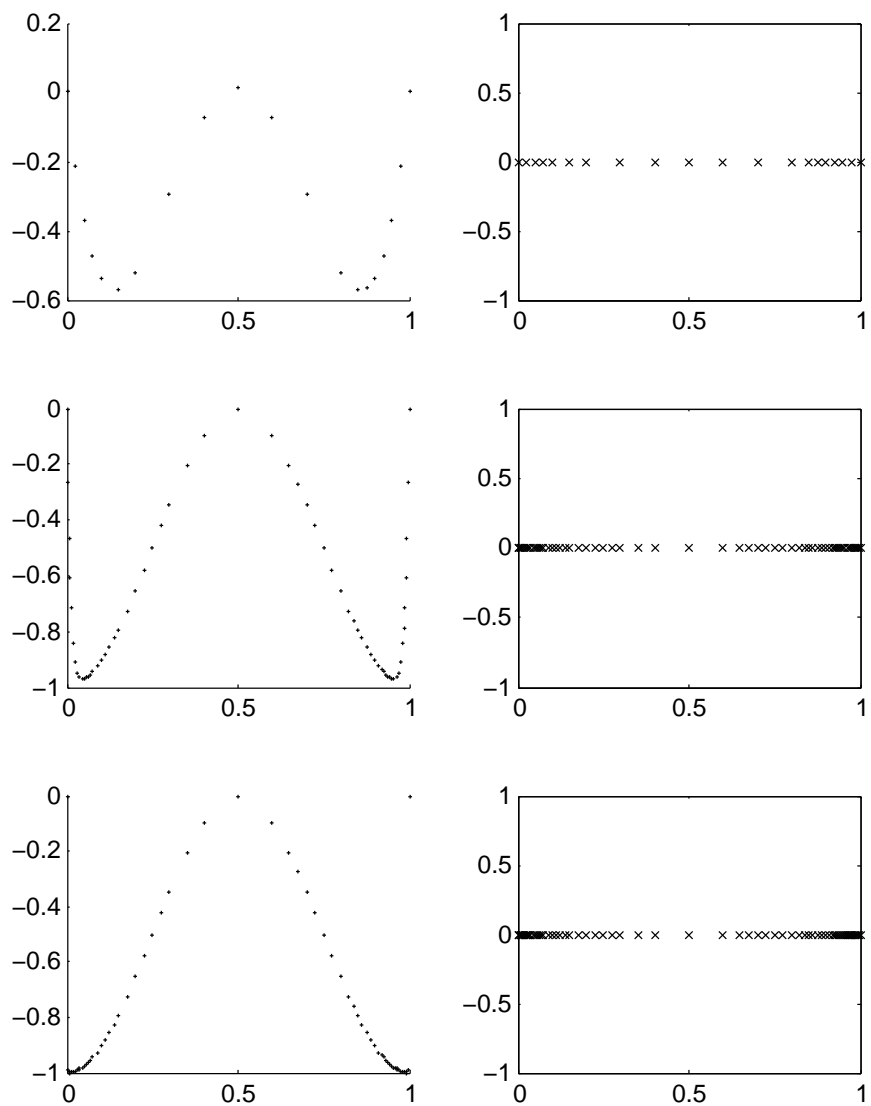
$\theta$	$\epsilon$	$\ u - u^J\ _\infty$	$\ u - u^J\ _2$	Korišćeno tačaka
$\theta = 10^{-2}$	$\epsilon = 10^{-1}$	$1.30002 * 10^{-5}$	$5.20838 * 10^{-7}$	321
		$3.15416 * 10^{-3}$	$4.28391 * 10^{-4}$	11
$\theta = 10^{-4}$	$\epsilon = 10^{-1}$	$1.48043 * 10^{-3}$	$1.59029 * 10^{-5}$	321
		$6.67414 * 10^{-3}$	$3.66009 * 10^{-4}$	25
$\theta = 10^{-7}$	$\epsilon = 10^{-1}$	$9.98442 * 10^{-3}$	$4.39901 * 10^{-5}$	321
		$7.99314 * 10^{-3}$	$3.17962 * 10^{-4}$	35
$\theta = 10^{-2}$	$\epsilon = 10^{-3}$	$1.3002 * 10^{-5}$	$5.20838 * 10^{-7}$	321
		$1.47301 * 10^{-4}$	$1.41668 * 10^{-5}$	20
$\theta = 10^{-4}$	$\epsilon = 10^{-3}$	$1.48043810^{-3}$	$1.59029 * 10^{-5}$	321
		$1.6564 * 10^{-5}$	$5.03611 * 10^{-7}$	63
$\theta = 10^{-7}$	$\epsilon = 10^{-3}$	$9.98442 * 10^{-3}$	$4.39901 * 10^{-5}$	321
		$7.68936 * 10^{-3}$	$1.53673 * 10^{-4}$	71

U prvoj vrsti tabele 1. date su vrednosti euklidske i beskonačne norme greške i broj korišćenih tačaka, za  $\epsilon = 0.1$  i  $\theta = 0.01$ . Primećujemo da je greška manja za  $N=1$ , odnosno algoritam kao da nije dao nikakav napredak. To se dešava zbog veličine praga kompresije  $\epsilon$ . Ukoliko smanjimo  $\epsilon$  dobijamo tačnije rešenje, što se vidi u četvrtoj vrsti. Prednost ove metode je u slučajevima kada imamo nagle promene. U tom slučaju je rešenje bolje ili malo lošije, ali uz korišćenje daleko manjeg broja tačaka. Rešenje možemo popraviti smanjivanjem parametra  $\epsilon$ .

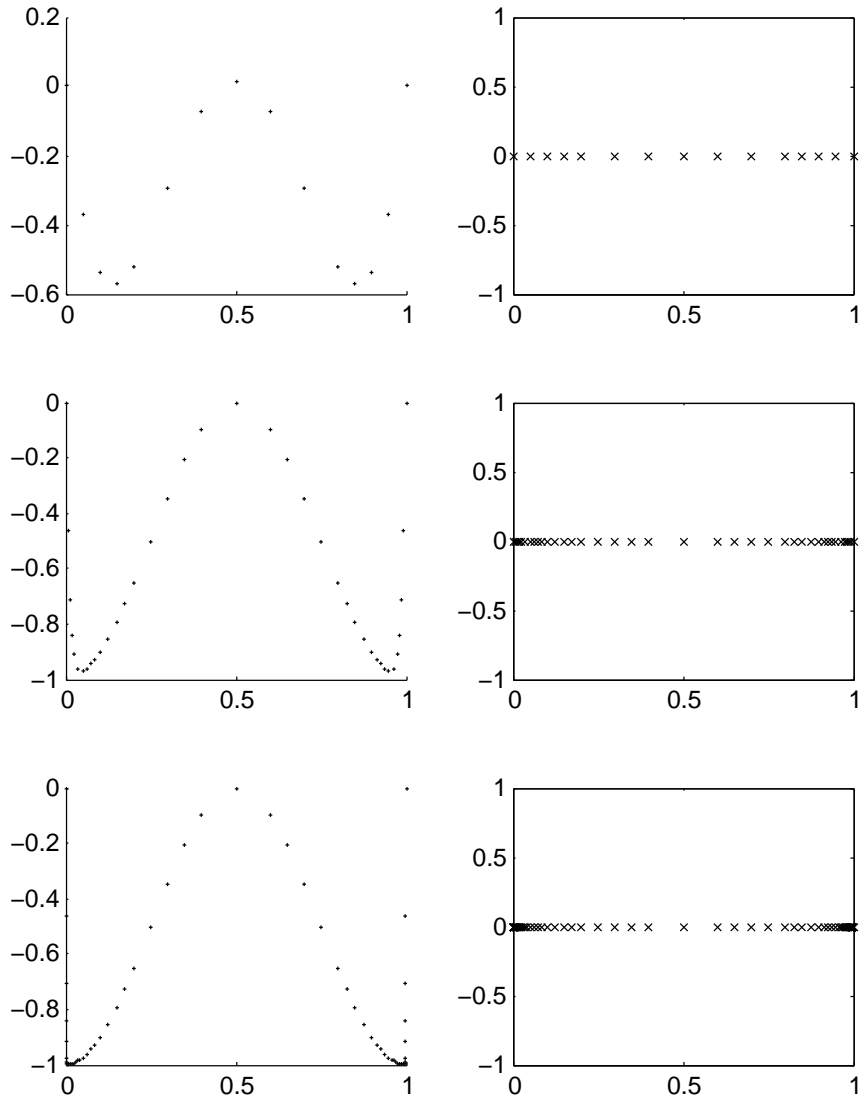
Testirajmo sada drugi algoritam na ovom primeru.

Tabela 2. Rezultati drugog algoritma na primeru

$\theta$	$\epsilon$	$\ u - u^J\ _\infty$	$\ u - u^J\ _2$	Korišćeno tačaka
$\theta = 10^{-2}$	$\epsilon = 10^{-1}$	$3.15416 * 10^{-3}$	$4.28391 * 10^{-4}$	11
$\theta = 10^{-4}$	$\epsilon = 10^{-1}$	$2.37693 * 10^{-2}$	$2.24488 * 10^{-3}$	15
$\theta = 10^{-7}$	$\epsilon = 10^{-1}$	$2.4062 * 10^{-2}$	$9.73791 * 10^{-4}$	35
$\theta = 10^{-2}$	$\epsilon = 10^{-3}$	$1.2954 * 10^{-4}$	$2.02387 * 10^{-5}$	15
$\theta = 10^{-4}$	$\epsilon = 10^{-3}$	$3.99244 * 10^{-4}$	$1.77999 * 10^{-5}$	39
$\theta = 10^{-7}$	$\epsilon = 10^{-3}$	$3.76139 * 10^{-4}$	$8.34321 * 10^{-6}$	79



Slika 3.2: Rešenja i raspored tačaka kolokacije pri korišćenju prvog algoritma na prvom primeru za  $\epsilon = 10^{-3}$  i redom  $\theta = 10^{-2}, 10^{-4}, 10^{-7}$



Slika 3.3: Rešenja i raspored tačaka kolokacije pri korišćenju drugog algoritma na prvom primeru za  $\epsilon = 10^{-3}$  i redom  $\theta = 10^{-2}, 10^{-4}, 10^{-7}$

Prednost drugog algoritma u odnosu na prvi, je što ne zadajemo nivo multirezolucije. Algoritam sam dodaje nivoe multirezolucije do trenutka kada više neme potrebe za tim, a to je onda kada su svi koeficijenti  $d_{j,k}$  manji od  $\epsilon$ , tada stajemo. Dalje, mana prvog algoritma je i što je moguće „promašiti” singularitet usled nedovoljnog broja nivoa multirezolucije.

Tabela 3. Primer loše izabranog broja multirezolucijskog nivoa

$\theta$	$\epsilon$	$\ u - u^J\ _\infty$	$\ u - u^J\ _2$	Korišćeno tačaka
$\theta = 10^{-7}$ J=5	$\epsilon = 10^{-1}$	$4,39901 * 10^{-5}$	$9.98442 * 10^{-3}$	321
		$3.17962 * 10^{-4}$	$7.99314 * 10^{-3}$	35
$\theta = 10^{-7}$ J=6	$\epsilon = 10^{-1}$	$3.0766 * 10^{-2}$	$6.79465 * 10^{-5}$	641
		$2.4062 * 10^{-2}$	$8.35388 * 10^{-4}$	40

U tabeli 3. primetno je da je greška manja, za manji broj multirezolucijskih nivoa, a to se dešava usled toga što mreža nije dovoljno gusta pa ne prepoznaje singularitet. S druge strane, drugi algoritam sam dodaje multirezolucijske nivoe, tako da se to neće desiti. Takodje, drugi algoritam će dati rezultat sa istom greškom u beskonačnoj normi, ali uz korišćenje manjeg broja tačaka. Razlog tome je što drugi algoritam koristi više tačaka koje su blizu singulariteta.

Drugi primer na kome ćemo testirati naše algoritme je:

$$\theta u'' + u' = 0,$$

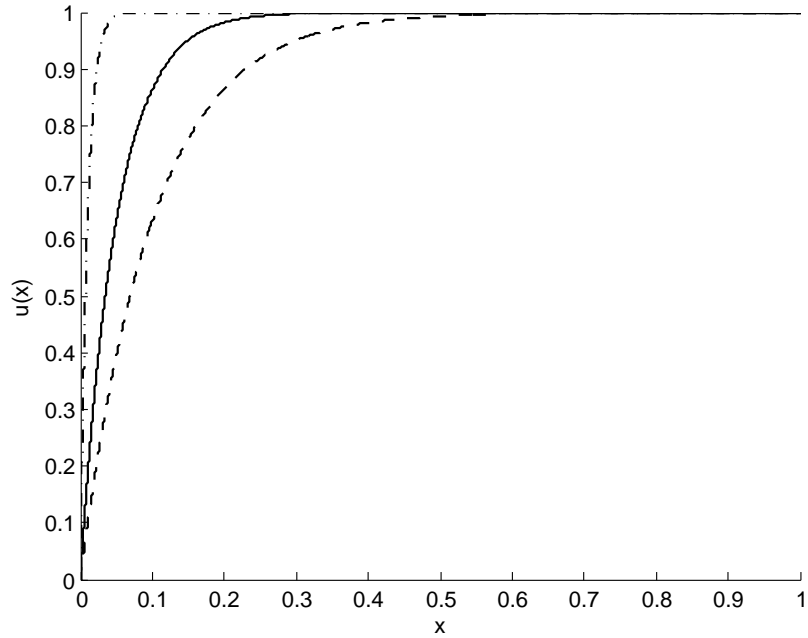
uz granične uslove:

$$u(0) = 0 \quad u(1) = 1.$$

Analitičko rešenje ovog problema je:

$$u = \frac{1 - e^{-\frac{x}{\epsilon}}}{1 - e^{-\frac{1}{\epsilon}}}.$$

Testirajmo prvi algoritam za  $N = 3$ , početni korak 0.1 i 5 nivoa multirezolucije.

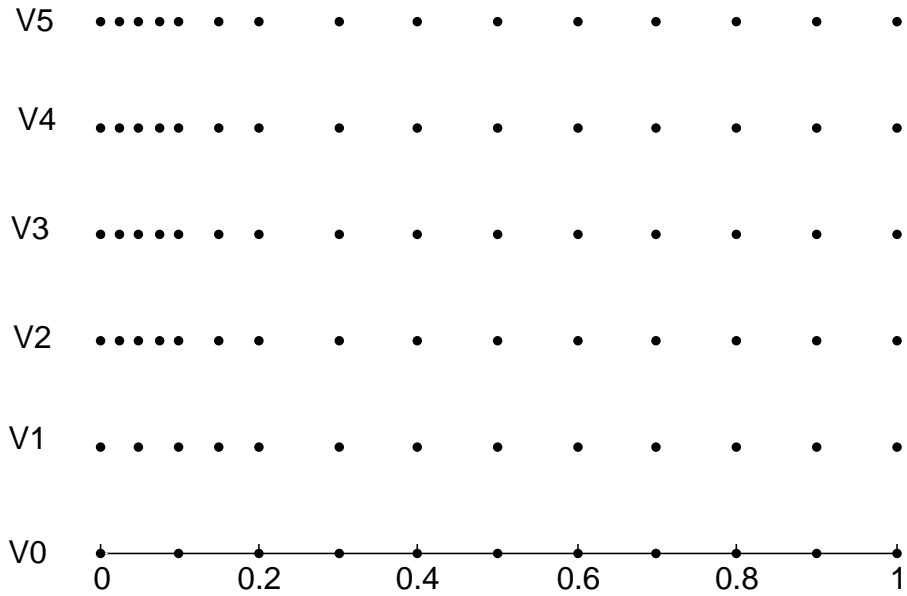


Slika 3.4: Analitičko rešenje primera 2. pri izboru  $\theta = 0.1$ (--),  $\theta = 0.05$ (-),  $\theta = 0.01$ (-.)

Tabela 4. Rezultati prvog algoritma na drugom primeru

$\theta$	$\epsilon$	$\ u - u^J\ _\infty$	$\ u - u^J\ _2$	Korišćeno tačaka
$\theta = 0.1$	$\epsilon = 10^{-2}$	$1.00579 * 10^{-4}$	$5.07338 * 10^{-6}$	321
		$1.70318 * 10^{-3}$	$1.67162 * 10^{-4}$	11
$\theta = 0.05$	$\epsilon = 10^{-2}$	$1.19806 * 10^{-4}$	$2.02894 * 10^{-6}$	321
		$5.24343 * 10^{-3}$	$7.30056 * 10^{-4}$	15
$\theta = 0.01$	$\epsilon = 10^{-2}$	$3.02055 * 10^{-3}$	$2.28649 * 10^{-5}$	321
		$1.96986 * 10^{-3}$	$1.70886 * 10^{-4}$	24
$\theta = 0.1$	$\epsilon = 10^{-5}$	$1.00579 * 10^{-4}$	$5.07338 * 10^{-6}$	321
		$1.22559 * 10^{-4}$	$1.68719 * 10^{-5}$	36
$\theta = 0.05$	$\epsilon = 10^{-5}$	$1.19806 * 10^{-4}$	$2.02894 * 10^{-6}$	321
		$6.22354 * 10^{-6}$	$7.86083 * 10^{-7}$	45
$\theta = 0.01$	$\epsilon = 10^{-5}$	$3.02055 * 10^{-3}$	$2.28649 * 10^{-5}$	321
		$6.21193 * 10^{-6}$	$1.40872 * 10^{-7}$	54

U tabeli 5. dati su rezultati drugog algoritma na drugom primeru sa početnim korakom  $h=0.1$  i  $N=3$



Slika 3.5: Multirezolucijski nivoi pri loše izabranom J

Tabela 5. Rezultati drugog algoritma na primeru

$\theta$	$\epsilon$	$\ u - u^J\ _\infty$	$\ u - u^J\ _2$	Korišćeno tačaka
$\theta = 0.1$	$\epsilon = 10^{-2}$	$1.70318 * 10^{-3}$	$1.67162 * 10^{-4}$	11
$\theta = 0.05$	$\epsilon = 10^{-2}$	$5.29029 * 10^{-3}$	$6.73944 * 10^{-4}$	13
$\theta = 0.01$	$\epsilon = 10^{-2}$	$1.93857 * 10^{-2}$	$3.63612 * 10^{-3}$	19
$\theta = 0.1$	$\epsilon = 10^{-5}$	$1.27999 * 10^{-4}$	$2.34348 * 10^{-5}$	22
$\theta = 0.05$	$\epsilon = 10^{-5}$	$2.39417 * 10^{-5}$	$2.64905 * 10^{-6}$	28
$\theta = 0.01$	$\epsilon = 10^{-5}$	$2.38367 * 10^{-5}$	$1.84331 * 10^{-6}$	38

Još jedna mana prvog algoritma može se uočiti ako pogledamo tačke koje se nalaze na multirezolucijskim nivoima, za izbor parametara dat u drugoj vrsti tabele (slika 3.5).

Vidimo da su nivoi  $V_i, 2 \leq i \leq 5$  identični, tako da nemamo potrebu za 5 nivoa. Ukoliko se testira ovaj problem za  $J=2$ , dobija se ista greška kao i za  $J=5$ , ali sa daleko manje izračunavanja.

Treći test primer je:

$$\theta u'' + 2xu' = 0,$$

uz granične uslove:

$$u(-1) = -1 \quad u(1) = 2.$$

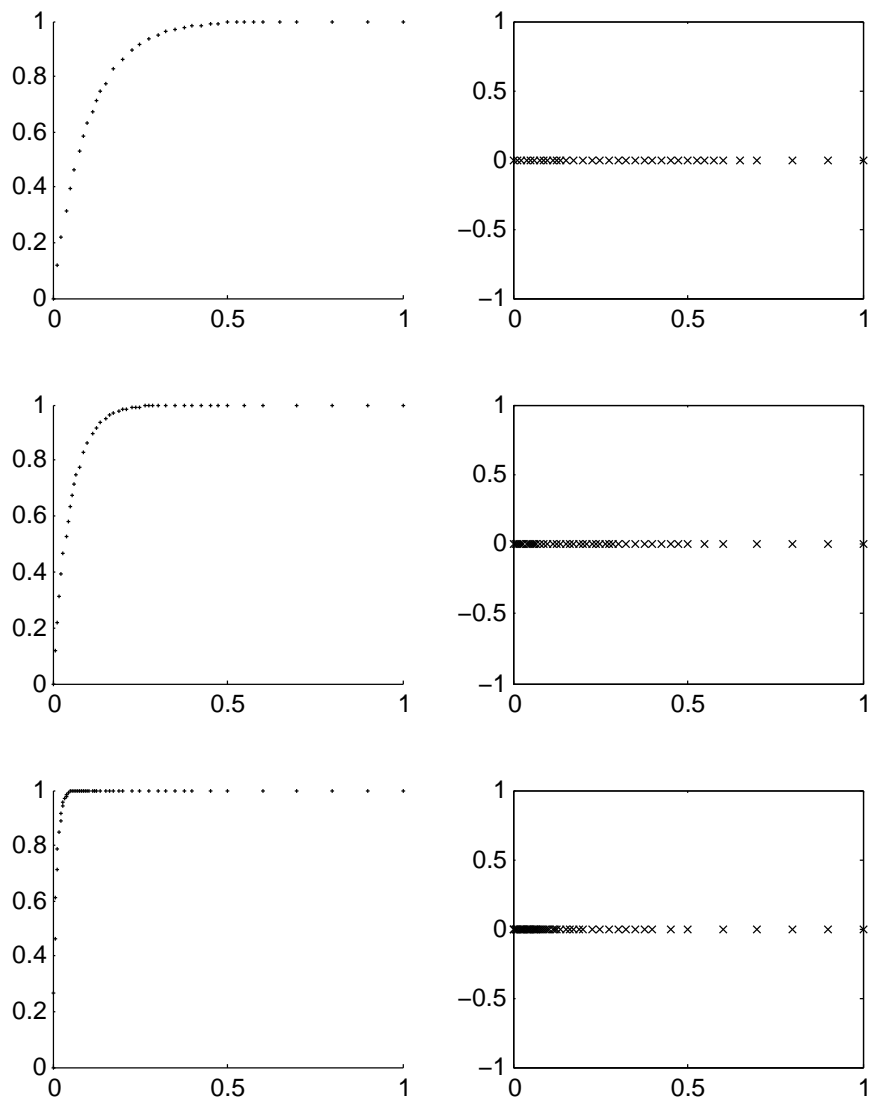
Analitičko rešenje ovog primera je:

$$u(t) = 3 \frac{\int_x^{-1} e^{-\frac{t^2}{\theta}} dt}{\int_1^{-1} e^{-\frac{t^2}{\theta}} dt} - 1.$$

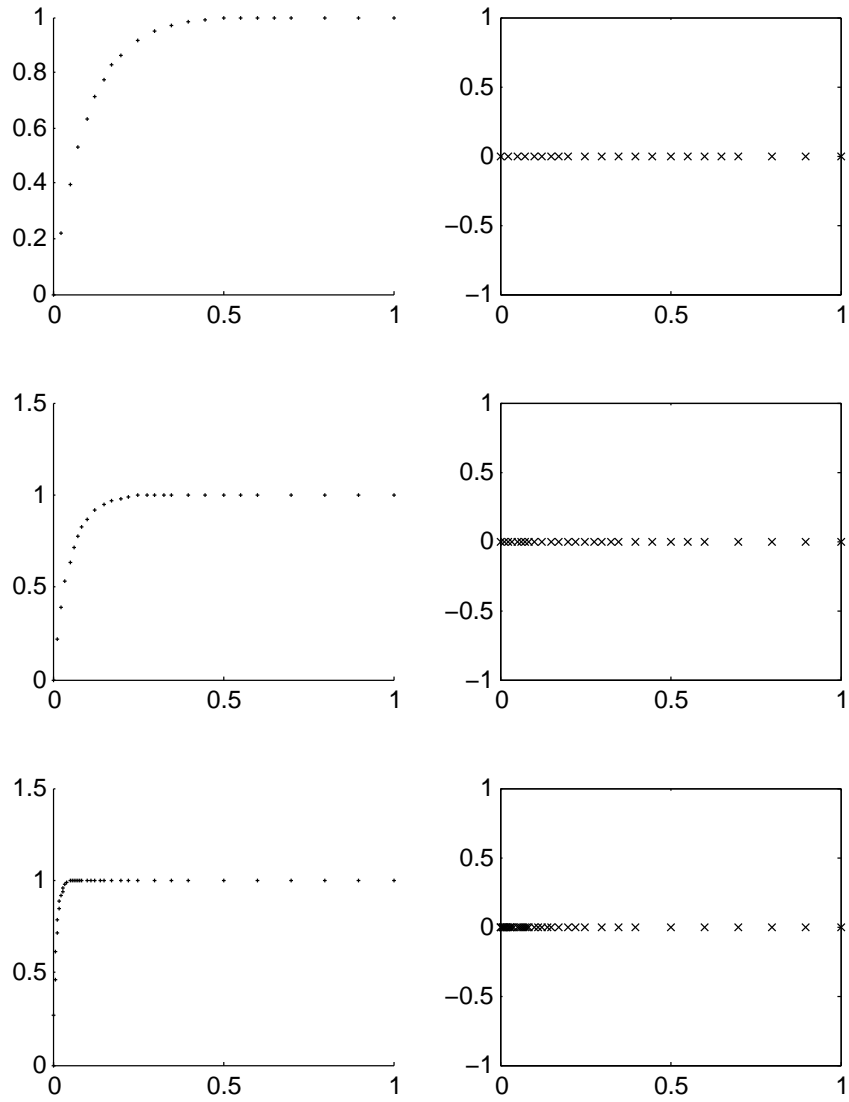
Ovaj primer predstavlja problem kada se singularitet nalazi na sredini intervala (slika 3.8.).

Algoritme testiramo za parametre  $N=2$ , početni korak 0.1,  $\epsilon = 0.00001$ ,  $\theta = 0.1, 0.01, 0.0001$ . U prvom algoritmu broj nivoa multirezolucije je  $J=5$ .

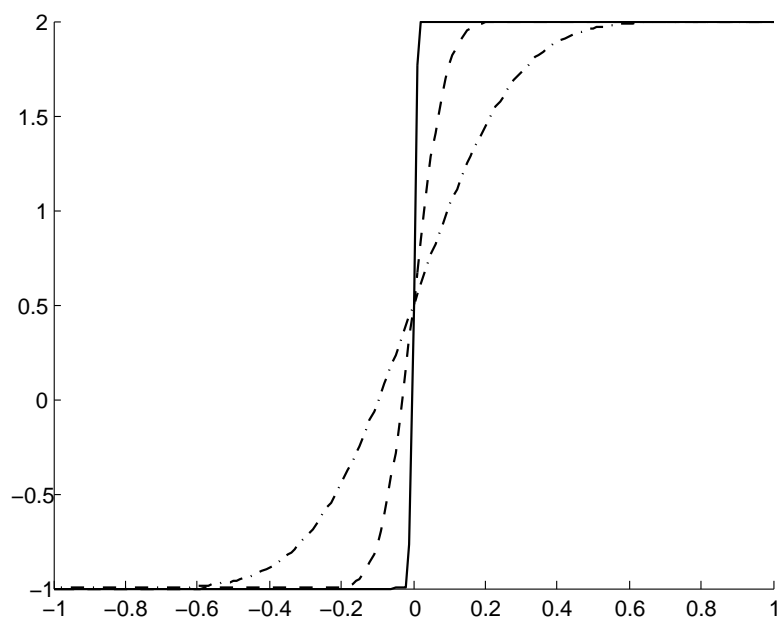
Ponovićemo da je tačnost rešenja uslovljena i tačnošću algoritma za rešavanja sistema linearnih jednačina, što je bitno napomenuti usled mogućnosti da matrica sistema bude skoro singularna za male vrednosti parametra  $\theta$ .



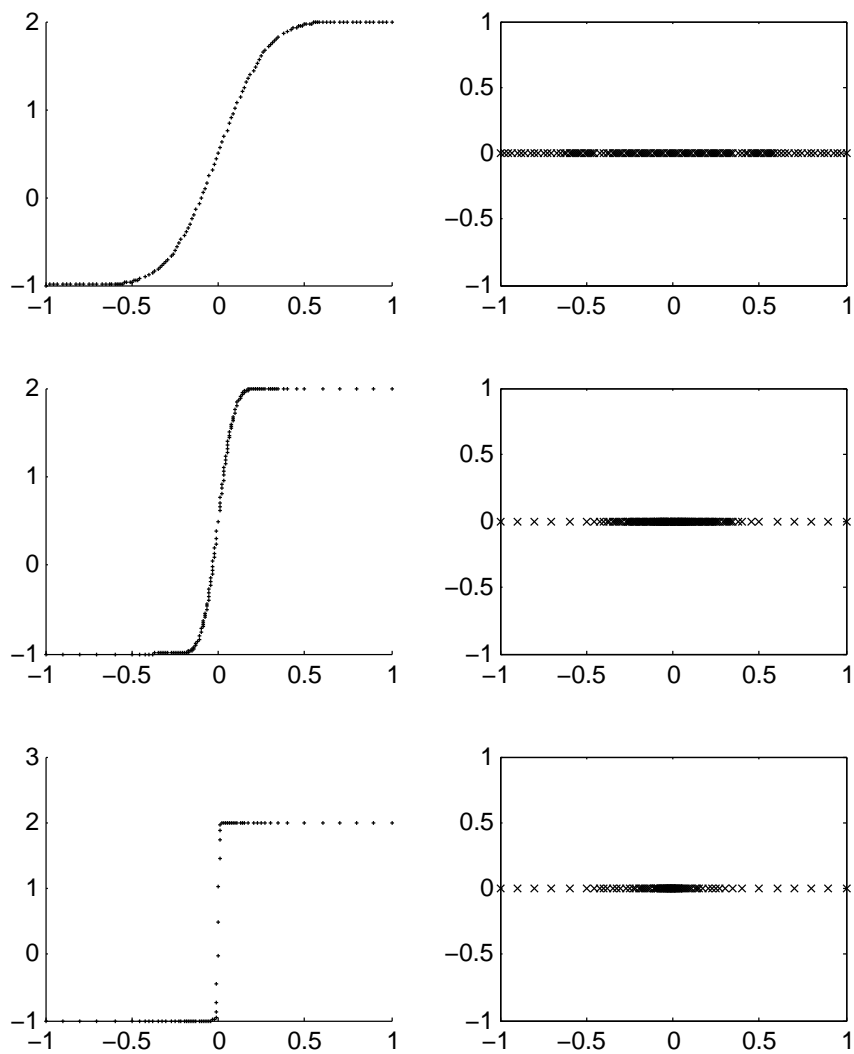
Slika 3.6: Rešenje i raspored tačaka pri korišćenju prvog algoritma na drugom primeru za  $\epsilon = 10^{-5}$ , i redom  $\theta = 0.1, 0.05, 0.01$



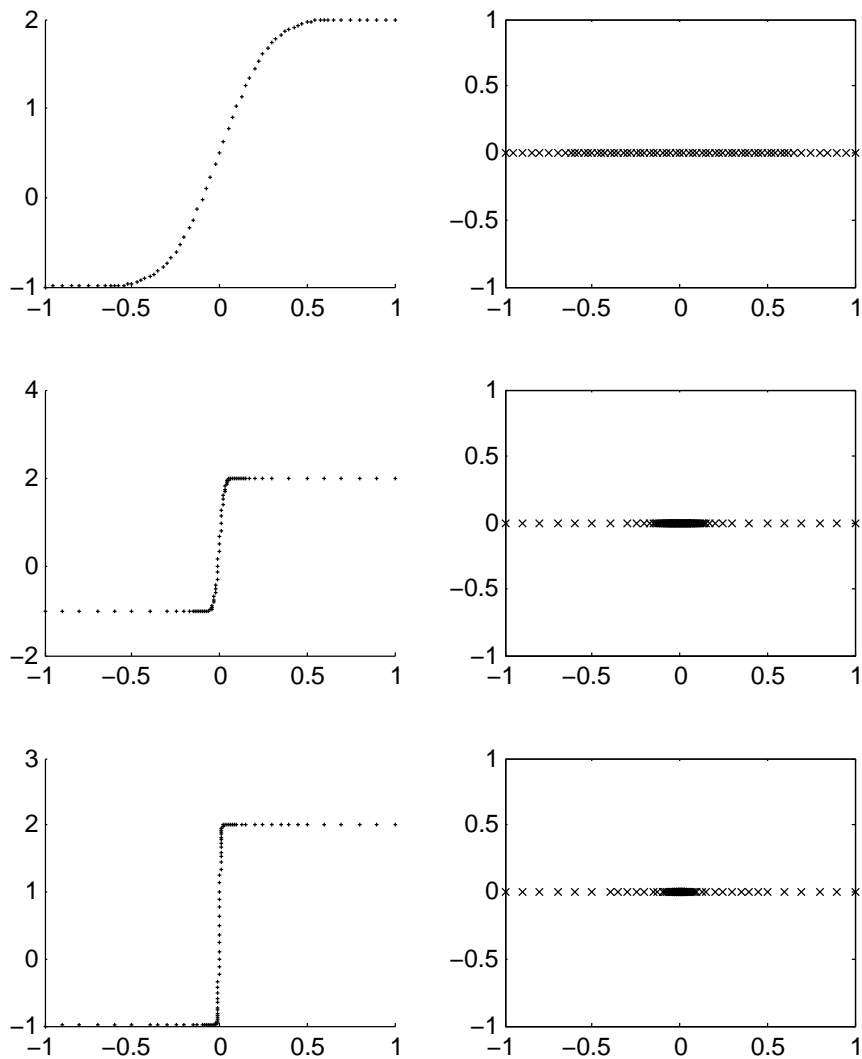
Slika 3.7: Rešenje i raspored tačaka pri korišćenju drugog algoritma na drugom primeru za  $\epsilon = 10^{-5}$ , i redom  $\theta = 0.1, 0.05, 0.01$



Slika 3.8: Analitičko rešenje primera 3. ukoliko je  $\theta = 0.1(-.)$ ,  $\theta = 0.01(- -)$ ,  $\theta = 0.0001(- \cdot)$



Slika 3.9: Rešenja i raspored tačaka kolokacije pri korišćenju prvog algoritma na trećem primeru za  $\epsilon = 0.00001$  i redom  $\theta=0.1$  (-.),  $\theta=0.01$ (- -),  $\theta=0.0001$ (-)



Slika 3.10: Rešenja i raspored tačaka kolokacije pri korišćenju drugog algoritma na trećem primeru za  $\epsilon = 0.00001$  i redom  $\theta=0.1$  (-.),  $\theta=0.01$ (- -),  $\theta=0.0001$ (-)

## 4

# Zaključak

U ovom radu predstavljeni su algoritmi za rešavanje diferencijalnih jednačina (DJ) korišćenjem talasića. Algoritmi su testirani na singularno-perturbovanim problemima, ili problemima sa malim parametrom. Male vrednosti parametra dovode do toga da je gradijent rešenja veliki u uskoj oblasti. Standardne metode numeričke matematike daju slabe rezultate pri rešavanju definisanih problema DJ čija rešenja imaju velike i nagle promene.

Dekompozicija talasićima je iskorišćena za adaptaciju mreže i interpolaciju, dok su podeljene razlike korišćene za aproksimaciju izvoda. Rezultati prikazani u poglavlju 3.3, pokazuju da se korišćenjem algoritama predstavljenih u poglavlju 3.1. vrši efikasna adaptacija mreže.

Vasiljev je u radovima [18]-[23] dao algoritam za rešavanje parcijalnih diferencijalnih jednačina. Prvi algoritam opisan u ovom radu predstavlja redukciju tog algoritma. Ideja ovakvog pristupa je da se krene od najfinijeg nivoa multirezolucije i da se iz mreže izbacuju tačke povezane sa talasićima, čiji su koeficijenti manji od nekog zadatog parametra.

Drugi algoritam, predstavljen u ovom radu, kreće od najgrubljeg nivoa multirezolucije i dodaje nove tačke, koje predstavljaju potpuno nove multirezolucijske nivoe.

Algoritme je moguće prilagoditi rešavanju parcijalnih diferencijalnih jednačina uvodjenjem dodatnog koraka koji bi predstavljao strategiju za predviđanje na vremenskim nivoima. U radovima [18]-[23] to je i uradjeno za prvi algoritam predstavljen u ovom radu. Po sličnom principu, verovatno je moguće izmeniti i drugi algoritam, što bi neka naredna istraživanja trebalo da pokažu.

Veliki problem u radu sa talasićima predstavlja rad na intervalima i neravnomernim mrežama. U ovom radu ti problem su rešeni korišćenjem druge generacije interpolacionih talasića. Lifting šema je iskorišćena za konstrukciju druge generacije talasića.

Naredna istraživanja treba usmeriti ka implementaciji WOFD (Wavelet Optimized Finite Difference Method) [5], [6] u algoritme. WOFD ima dobru osobinu što nije potrebno zadavati  $N$ , već ga WOFD na osnovu koeficijenata talasića sam određuje. Na taj način izbegava se uticaj Rungeovog efekta, da je funkcija male varijacije bolje aproksimirana polinomom nižeg stepena. Konkretno, WOFD aproksimira funkciju polinomima višeg reda u oblastima gde postoje velike i nagle promene, a u oblastima bez većih promena koristi polinome nižeg stepena.

Mana ovih metoda je nedostatak efikasne ocene greške i ocene brzine konvergencije. Ovo su problemi koje bi trebalo neka buduća istraživanja da reše.

# Literatura

- [1] BACRY E., MALLAT S., PAPANICOLAOU G., *A wavelet based space-time adaptive numerical method for partial differential equations*, RAIRO Model. Math. Anal. Numer., 26, 793 (1992).
- [2] DONOHO D. L., *Interpolating Wavelet Transforms*, Technical Report 408, Department of Statistics, Stanford University, 1992.
- [3] HOLMSTROM M., WALDEN J., *Adaptive wavelet methods for hyperbolic PDEs*, J. Sci. Comput., 13, 19 (1998).
- [4] HOLMSTROM M., *Solving hyperbolic PDEs using interpolating wavelets*, SIAM J. Sci. Comput. 21, 405 (1999).
- [5] JAMESON L., *A Wavelet-optimized, Very High Order Adaptive Grid And Order Numerical Method*, SIAM J. Sci. Comput. 19, 1980 (1998).
- [6] JAMESON L., *A Wavelet-based Grid Generation*, NASA Contractor Report, ICASE Report No. 96-59 (1996).
- [7] JOVANOVIĆ B., RADUNOVIĆ D., *Numerička analiza*, Matematički fakultet, Beograd, 2003.
- [8] KUMAR V., MEHRA M., *Wavelet optimized finite difference method using interpolating wavelets for solving singularly perturbed problems*, Journal of Wavelet Theory and Applications, ISSN 0973-6336, Volume 1, Number 1 (2007), pp. 8396
- [9] RADUNOVIĆ D., *Talasići (Wavelets)*, Akademska misao, Beograd, 2005.
- [10] RADUNOVIĆ D., SAMARDŽIĆ A., MARIĆ F., *Numeričke metode (zbirka zadataka kroz C, Fortran i Matlab)*, Akademska misao, Beograd, 2005.
- [11] RADUNOVIĆ D., *Numeričke metode*, Akademska misao, Beograd, 2004.

- [12] SANTOS J.C., CRUZ P., ALVES M.A, OLIVEIRA P.J., MAGALHAES F.D., MENDES A., *Adaptive multiresolution approach for two-dimensional PDEs*, Comput. Methods Appl. Mech. Engrg. 193 (2004) 405.
- [13] SWELDENS W., *The lifting scheme: A construction of second generation wavelets*, SIAM J. Math. Anal. 29, 511 (1998).
- [14] SWELDENS W., SCHRÖEDER P., *Building Your Own Wavelets at Home*, “Wavelets in Computer Graphics”, ACM SIGGRAPH Course Notes, 1996.
- [15] SWELDENS W., *The Construction and Application of Wavelets in Numerical Analysis*, Ph.D. Thesis, 1995,
- [16] UDOVIČIĆ Z., *Odredjivanje praga kompresije pri transformaciji ortonormiranim talasićima*, Magistarski rad, (Matematički fakultet, Beograd).
- [17] VASILYEV O. V., BOWMAN C., *Second-Generation Wavelet Collocation Method for the Solution of Partial Differential Equations*, J. Comput. Phys. 165, 660-693 (2000).
- [18] VASILYEV O. V., PAOLUCCI S., *A fast adaptive wavelet collocation algorithm for multidimensional PDEs*, J. Comput. Phys. 138, 16 (1997).
- [19] VASILYEV O. V., PAOLUCCI S., *A dynamically adaptive multilevel wavelet collocation method for solving partial differential equations in a finite domain*, J. Comput. Phys. 125, 498 (1996).
- [20] VASILYEV O. V., PAOLUCCI S., SEN M., *A multilevel wavelet collocation method for solving partial differential equations in a finite domain*, J. Comput. Phys. 120, 33 (1995).
- [21] WALDEN J., *Filter bank methods for hyperbolic PDEs*, SIAM J. Numer. Anal. 36, 1183 (1999).
- [22] VASILYEV O. V., KEVLAHAN N. K. R., *An adaptive multilevel wavelet collocation method for elliptic problems*, Journal of Computational Physics 206, 412 (2005).
- [23] VASILYEV O.V., KEVLAHAN N. K. R., ALAM J. M *Simultaneous spacetime adaptive wavelet solution of nonlinear parabolic differential equations*, Journal of Computational Physics 214, 829 (2006).

- [24] WALDEN J., *Filter bank methods for hyperbolic PDEs*, SIAM J. Numer. Anal. 36, 1183 (1999).