

# Wpływ zaburzeń współczynników na wartość kombinacji liniowej wielomianów Czebyszewa (**zad. P3.9**)

Paweł Laskoś

grupa dr. Pawła Woźnego, nr indeksu 169827

9 stycznia 2006

## Spis treści

<b>1</b>	<b>Wstęp</b>	<b>1</b>
<b>2</b>	<b>Opis teoretyczny</b>	<b>2</b>
2.1	Współczynniki $J_n, K_{n-1}$ . . . . .	2
2.2	Współczynniki $I_n$ . . . . .	4
2.3	Algorytm Clenshawa . . . . .	4
2.4	Zaburzenia . . . . .	4
<b>3</b>	<b>Wyniki doświadczenia</b>	<b>5</b>
<b>4</b>	<b>Wnioski</b>	<b>5</b>

## 1 Wstęp

Celem eksperymentu było sprawdzenie wpływu małych zaburzeń  $c_k$  na wartość wielomianu

$$s_n(x) = \sum_{k=0}^n c_k T_k(x) \quad (1)$$

w punkcie  $x$ , przy zaburzaniu wartości stałych  $c_k$ . Obliczenia zrealizowano dla  $s_n = I_n, J_n, K_{n-1}$ , gdzie  $I_n$  to wielomian interpolujący funkcję  $f$  w węzłach

$$t_{n+1,k} = \cos \frac{2k-1}{2n+2} \pi, \quad (2)$$

będących zerami  $n+1$ -szego wielomianu Czebyszewa (pierwszego rodzaju), wyrażający się wzorem

$$I_n(x) = \frac{2}{n+1} \sum_{j=0}^n \left( \sum_{k=1}^{n+1} f(t_{n+1,k}) T_j(t_{n+1,k}) \right) T_j(x); \quad (3)$$

$J_n$  to wielomian interpolujący funkcję  $f$  w węzłach

$$u_{n-1,k} = \cos \frac{k}{n} \pi, \quad (4)$$

będących zerami  $n - 1$ -szego wielomianu Czebyszewa (drugiego rodzaju), wyrażający się wzorem

$$J_n(x) = \frac{2}{n} \sum_{j=0}^n \left( \sum_{k=0}^n f(u_{n-1,k}) T_j(u_{n-1,k}) \right) T_j(x); \quad (5)$$

zaś  $K_{n-1}$  to  $n - 1$ -szy wielomian optymalny w sensie aproksymacji jednostajnej dla funkcji  $f$  na zbiorze

$$\{u_{n-1,0}, u_{n-1,1}, \dots, u_{n-1,n}\}, \quad (6)$$

wyrażający się wzorem

$$K_n(x) = \frac{2}{n+1} \sum_{j=0}^{n-1} \left( \sum_{k=0}^n f(u_{n-1,k}) T_j(u_{n-1,k}) \right) T_j(x). \quad (7)$$

Za  $f$  przyjęto funkcję wykładniczą  $e^x$ , trygonometryczną  $\sin x$  oraz 11 wielomian Czebyszewa  $T_{11}(x)$ .

## 2 Opis teoretyczny

W [2] znaleźć można (odpowiednio na stronach 89, 94 i 101) dowody poprawności wyżej przytoczonych postaci wielomianów  $I_n, J_n, K_{n-1}$ .

### 2.1 Współczynniki $J_n, K_{n-1}$

Współczynniki kombinacji liniowej wielomianów Czebyszewa dla  $J_n, K_{n-1}$  wyliczyć można następującym algorytmem, opisanym w [2, s. 283 i nast.]:

$$u_k := u_{n-1,k} = \cos \frac{k}{n} \pi, \quad (8)$$

$$f_k := f(u_k) \quad (k = 0, \dots, n), \quad (9)$$

$$s_k := \begin{cases} f_k + f_{n-k} & (k = 0, \dots, \lfloor \frac{n}{2} \rfloor), \\ f_{\frac{n}{2}} & (2 \mid n, k = \frac{n}{2}), \end{cases} \quad (10)$$

$$d_k := \begin{cases} f_k - f_{n-k} & (k = 0, \dots, \lfloor \frac{n}{2} \rfloor), \\ f_{\frac{n}{2}} & (2 \mid n, k = \frac{n}{2}), \end{cases} \quad (11)$$

$$\alpha_j := \frac{2}{n} \times \begin{cases} \sum_{l=0}^{\lfloor \frac{n}{4} \rfloor} s_{2l} T_l(u_{2j}) + \sum_{l=0}^{\lfloor \frac{n-2}{4} \rfloor} s_{2l+1} T_{2l+1}(u_j) & (2 \mid j), \\ \sum_{l=0}^{\lfloor \frac{n}{4} \rfloor} d_{2l} T_l(u_{2j}) + \sum_{l=0}^{\lfloor \frac{n-2}{4} \rfloor} d_{2l+1} T_{2l+1}(u_j) & (2 \nmid j), \end{cases} \quad (12)$$

$$\alpha_{n-j} := \frac{2}{n} \times \begin{cases} \sum_{l=0}^{\lfloor \frac{n}{4} \rfloor} s_{2l} T_l(u_{2j}) - \sum_{l=0}^{\lfloor \frac{n-2}{4} \rfloor} s_{2l+1} T_{2l+1}(u_j) & (2 \mid n-j), \\ \sum_{l=0}^{\lfloor \frac{n}{4} \rfloor} d_{2l} T_l(u_{2j}) - \sum_{l=0}^{\lfloor \frac{n-2}{4} \rfloor} d_{2l+1} T_{2l+1}(u_j) & (2 \nmid n-j), \end{cases} \quad (13)$$

$$\alpha_{\frac{n}{2}} := \frac{2}{n} \times \begin{cases} \sum_{l=0}^{\lfloor \frac{n}{4} \rfloor} s_{2l} T_l(-1) & (2 \mid \frac{n}{2}), \\ \sum_{l=0}^{\lfloor \frac{n}{4} \rfloor} d_{2l} T_l(-1) & (2 \nmid \frac{n}{2}). \end{cases} \quad (14)$$

Wartości (12,13) wyliczamy dla  $j = 0, \dots, \lfloor \frac{n-1}{2} \rfloor$ , zaś wartość (14) dla  $2 \mid n$ . Wtedy mamy

$$J_n(x) = \sum_{j=0}^n \alpha_j T_j(x), \quad (15)$$

$$K_{n-1}(x) = \sum_{j=0}^{n-1} \alpha_j T_j(x). \quad (16)$$

Dowód poprawności algorytmu, czyli tożsamości

$$\frac{n}{2} \alpha_j = \sum_{k=0}^n f(u_k) T_j(u_k), \quad (17)$$

przytoczymy za [2]:

*Dowód.* Ponieważ  $T_j(u_{n-k}) = T_j(-u_k) = (-1)^j T_j(u_k)$ , to

$$\frac{n}{2} \alpha_j = \frac{1}{2} (f_0 + (-1)^j f_n) T_j(u_0) + (f_1 + (-1)^j f_{n-1}) T_j(u_1) + \dots, \quad (18)$$

gdzie ostatnim składnikiem jest  $f_{\frac{n}{2}} T_j(u_{\frac{n}{2}})$  dla  $n$  parzystych i  $(f_{\frac{n-1}{2}} + (-1)^j f_{\frac{n+1}{2}}) T_j(u_{\frac{n-1}{2}})$  dla  $n$  nieparzystych. Wobec tego, stosując wzory (10,11) oraz tożsamość  $T_l(u_{nk}) = T_k(u_{nl})$ , mamy

$$\frac{n}{2} \alpha_j = \begin{cases} \sum_{k=0}^{\lfloor \frac{n}{2} \rfloor} s_k T_k(u_j) & (2 \mid j), \\ \sum_{k=0}^{\lfloor \frac{n}{2} \rfloor} d_k T_k(u_j) & (2 \nmid j). \end{cases} \quad (19)$$

Jeśli  $n$  jest parzyste i  $j = \frac{n}{2}$ , to  $u_j = 0$  i w sumie (19) wartości  $T_1(u_j), T_3(u_j), \dots$  są zerami. Dlatego, jeśli na przykład  $\frac{n}{2}$  jest też parzyste, to

$$\frac{n}{2} \alpha_n = \sum_{l=0}^{\lfloor \frac{n}{4} \rfloor} s_{2l} T_{2l}(0) = \sum_{l=0}^{\lfloor \frac{n}{4} \rfloor} s_{2l} T_l(-1), \quad (20)$$

co uzasadnia wzór (14). Jeśli  $j < \frac{n}{2}$ , to sumę z (19) dzielimy na część zawierającą wielomiany  $T_0, T_2, \dots$ , równą na przykład (dla  $j$  parzystego)

$$\sum_{l=0}^{\lfloor \frac{n}{4} \rfloor} s_{2l} T_{2l}(u_j), \quad (21)$$

i na część zawierającą wielomiany  $T_1, T_3, \dots$ , równą w tymże przypadku

$$\sum_{l=0}^{\lfloor \frac{n-2}{4} \rfloor} s_{2l+1} T_{2l+1}(u_j). \quad (22)$$

Ponieważ  $T_{2l}(u_j) = T_l(u_{2j})$ , więc są prawdziwe wzory (12).

Dla  $j < \frac{n}{2}$  wartość  $\alpha_{n-j}$  obliczamy z wzoru wynikającego z (19):

$$\frac{n}{2} \alpha_{n-j} = \begin{cases} \sum_{k=0}^{\lfloor \frac{n}{2} \rfloor} s_k T_k(-u_j) & (2 \mid n-j), \\ \sum_{k=0}^{\lfloor \frac{n}{2} \rfloor} d_k T_k(-u_j) & (2 \nmid n-j). \end{cases} \quad (23)$$

Pierwsza z tych sum jest różnicą wyrażeń (21,22), więc wzory (13) są również poprawne.

Dla analogicznych, pominiętych przypadków przeprowadzić można identyczne rozumowanie.  $\square$

Najważniejszą zaletą przedstawionego algorytmu jest przyspieszenie obliczeń o stały czynnik (ok. 2). Zamiast obliczać wartości (w liniowym czasie, dzięki algorytmowi Clenshawa)  $n$  wielomianów stopnia  $n$ , obliczamy wartości około  $2n$  wielomianów stopnia ok.  $\frac{n}{4}$ .

## 2.2 Współczynniki $I_n$

W wyprowadzeniu powyższego algorytmu kluczowe jest przejście między sumami następujących postaci

$$\sum_k c_k T_j(u_k) = \sum_k c_k T_k(u_j), \quad (24)$$

co umożliwia zastosowanie algorytmu Clenshawa do prostego wyliczenia wartości współczynników. Jednak węzły  $t_{n+1,k}$  nie mają tej własności. Wyliczymy zatem współczynniki  $I_n$  wypełniając kwadratową tabelę liczbami  $t_{i,j} = c_i T_j(t_{n+1,i})$ , by potem zsumować jej kolumny.

Złożoność takiego podejścia jest rzędu  $n^2$ , podobnie jak w poprzednim przypadku, jednak ilość obliczeń jest większa.

## 2.3 Algorytm Clenshawa

Na końcu warto omówić algorytm obliczania wartości sum postaci  $S(x) = \sum_{k=0}^n c_k T_k(x)$ , określonego ogólnie dla ciągu wielomianów  $F_n$  spełniających warunki

$$F_{n+2}(x) = \alpha(n+1, x)F_{n+1}(x) + \beta(n+1, x)F_n(x), \quad (25)$$

$$F_1(x) = \alpha(0, x)F_0(x). \quad (26)$$

Z [3] wiemy, że po wyliczeniu ciągu wartości

$$y_{n+2} := y_{n+1} := 0, \quad (27)$$

$$y_k := \alpha(k, x)y_{k+1} + \beta(k+1, x)y_{k+2} + c_k, \quad (28)$$

otrzymamy

$$S(x) = c_0 F_0(x) + y_1 F_1(x) + \beta(1, x)F_0(x)y_2, \quad (29)$$

jednak korzystając z (26) łatwo napisać ostatecznie

$$S = y_0 F_0(x). \quad (30)$$

Algorytm ten realizowany będzie wyłącznie dla wielomianów Czebyszewa, dla których mamy

$$\alpha(0, x) = x, \quad (31)$$

$$\alpha(k, x) = 2x \quad (k = 1, 2, \dots) \quad (32)$$

$$\beta(k, x) = -1 \quad (k = 1, 2, \dots), \quad (33)$$

$$F_0(x) = 1, \quad (34)$$

czyli w tym przypadku po prostu  $S(x) = y_0$ .

## 2.4 Zaburzenia

Poza efektywnym wyliczeniem wartości współczynników  $s_n(x)$ , naszym zadaniem jest sprawdzenie zmian jego wartości po zaburzeniu współczynników o wylosowane wartości  $p_k \in [-\delta, \delta]$  dla małego  $\delta$ , czyli przeanalizowanie wartości sumy

$$\tilde{s}_n(x) = \sum_{k=0}^n (c_k + p_k) T_k(x). \quad (35)$$

Jeśli rozbijemy  $\tilde{s}_n$  na dwie sumy, przekonamy się, że zmiana wartości po zaburzeniu współczynników zależy wyłącznie od tych zaburzeń oraz indeksu  $n$  (i, oczywiście, wartości argumentu  $x$ , dla której ją wyliczymy).

## 3 Wyniki doświadczenia

Załączony kod `chebsum.c` implementuje w języku ANSI C omówione algorytmy dla  $n = 9$ , w arytmetyce podwójnej precyzji, obliczające wartości wielomianów w punktach  $t_{10,k}$  dla  $I_9$  oraz  $u_{8,k}$  dla  $J_9, K_8$ . Wylosowane zaburzenia (dla  $\delta = 0,001$ )  $p_k$  oraz wyniki obliczeń podano w tabelach 1, 2.

## 4 Wnioski

Analiza wyników potwierdza fakt, że różnica  $s_n - \tilde{s}_n$  zależy tylko od zaburzeń oraz  $n$  – różnice  $I_n - \tilde{I}_n$ ,  $J_n - \tilde{J}_n$ ,  $K_n - \tilde{K}_n$ , identyczne dla każdej funkcji interpolowanej (aproxymowanej)  $f$  zestawiono w tabeli 3. Wartości owych różnic są niewielkie, żadna z nich nie przekracza  $\delta$ , zatem można powiedzieć, że zadanie obliczania wartości wielomianu algorytmem Clenshawa jest dobrze uwarunkowane.

Warto jeszcze nadmienić, dlaczego wartości zebrane w tabeli 3 różnią się: otóż wyliczono je dla różnych górnych indeksów sumowania (dla  $I_n, J_n - 9$ , dla  $K_{n-1} - 8$ ), dla różnych argumentów  $x$  (dla  $I_n$  – węzły  $t_k$ , dla  $J_n, K_{n-1}$  – węzły  $u_k$ ), i ostatecznie – jak widać z kodu programu – dla różnych zaburzeń, gdyż  $p_k$  nie są zaburzeniami współczynników  $c_k$  w sensie wzoru

$$s_n(x) = \sum c_k T_k(x), \quad (36)$$

lecz wzoru

$$s_n(x) = \sum' c_k T_k(x) \quad (37)$$

dla  $I_n, K_{n-1}$  oraz

$$s_n(x) = \sum'' c_k T_k(x) \quad (38)$$

dla  $J_n$ .

## Literatura

- [1] Stanisław Lewanowicz, *Aproxymacja funkcji*, notatki do wykładu z analizy numerycznej, 2005<sup>1</sup>.

---

<sup>1</sup><http://www.ii.uni.wroc.pl/sle/an-apr.pdf>

Zaburzenia:

$p_9$	-0,0002112341
$p_8$	0,0005968801
$p_7$	-0,0006048973
$p_6$	0,0005364592
$p_5$	0,0001079399
$p_4$	0,0002577418
$p_3$	0,0000268018
$p_2$	0,0008323901
$p_1$	0,0004345939
$p_0$	0,0002139378

$f(x) = \sin x$

$x$	$I_9(x)$	$\tilde{I}_9(x)$
-0,9876883406	-0,8347553619	-0,8344783382
-0,8910065242	-0,7777048722	-0,7780018607
-0,7071067812	-0,6496369391	-0,6494747047
-0,4539904997	-0,4385552953	-0,4387103686
-0,1564344650	-0,1557972080	-0,1560111114
0,1564344650	0,1557972080	0,1557682952
0,4539904997	0,4385552953	0,4385365176
0,7071067812	0,6496369391	0,6496531476
0,8910065242	0,7777048722	0,7778112606
0,9876883406	0,8347553619	0,8351211004

$x$	$J_9(x)$	$\tilde{J}_9(x)$	$K_8(x)$	$\tilde{K}_8(x)$
-1,0000000000	-0,8414709848	-0,8409217362	-0,8414709743	-0,8409451961
-0,9396926208	-0,8073767737	-0,8076016131	-0,8073767842	-0,8075781532
-0,7660444431	-0,6932900635	-0,6932476435	-0,6932900530	-0,6932711034
-0,5000000000	-0,4794255386	-0,4794805856	-0,4794255491	-0,4794571256
-0,1736481777	-0,1727768036	-0,1730545626	-0,1727767931	-0,1730780225
0,1736481777	0,1727768036	0,1727690680	0,1727767931	0,1727925280
0,5000000000	0,4794255386	0,4793916621	0,4794255491	0,4793682021
0,7660444431	0,6932900635	0,6933357861	0,6932900530	0,6933592460
0,9396926208	0,8073767737	0,8075839508	0,8073767842	0,8075604908
1,0000000000	0,8414709848	0,8419574873	0,8414709743	0,8419809473

Tabela 1: Wyniki działania programu

$$f(x) = e^x$$

$x$	$I_9(x)$	$\tilde{I}_9(x)$
-0,9876883406	0,3724366433	0,3727136670
-0,8910065242	0,4102426258	0,4099456373
-0,7071067812	0,4930686914	0,4932309257
-0,4539904997	0,6350887667	0,6349336934
-0,1564344650	0,8551875605	0,8549736571
0,1564344650	1,1693341275	1,1693052148
0,4539904997	1,5745830385	1,5745642607
0,7071067812	2,0281149816	2,0281311902
0,8910065242	2,4375819021	2,4376882904
0,9876883406	2,6850204400	2,6853861785

$x$	$J_9(x)$	$\tilde{J}_9(x)$	$K_8(x)$	$\tilde{K}_8(x)$
-1,0000000000	0,3678794412	0,3684286898	0,3678794522	0,3684052304
-0,9396926208	0,3907479247	0,3905230852	0,3907479137	0,3905465447
-0,7660444431	0,4648481699	0,4648905899	0,4648481809	0,4648671305
-0,5000000000	0,6065306597	0,6064756127	0,6065306487	0,6064990721
-0,1736481777	0,8405925849	0,8403148259	0,8405925959	0,8402913665
0,1736481777	1,1896369513	1,1896292158	1,1896369403	1,1896526752
0,5000000000	1,6487212707	1,6486873942	1,6487212817	1,6486639347
0,7660444431	2,1512400496	2,1512857721	2,1512400385	2,1513092315
0,9396926208	2,5591946542	2,5594018313	2,5591946652	2,5593783719
1,0000000000	2,7182818285	2,7187683310	2,7182818174	2,7187917904

$$f(x) = T_{11}(x)$$

$x$	$I_9(x)$	$\tilde{I}_9(x)$
-0,9876883406	0,1564344650	0,1567114887
-0,8910065242	-0,4539904997	-0,4542874882
-0,7071067812	0,7071067812	0,7072690155
-0,4539904997	-0,8910065242	-0,8911615975
-0,1564344650	0,9876883406	0,9874744372
0,1564344650	-0,9876883406	-0,9877172533
0,4539904997	0,8910065242	0,8909877465
0,7071067812	-0,7071067812	-0,7070905727
0,8910065242	0,4539904997	0,4540968881
0,9876883406	-0,1564344650	-0,1560687266

$x$	$J_9(x)$	$\tilde{J}_9(x)$	$K_8(x)$	$\tilde{K}_8(x)$
-1,0000000000	-1,0000000000	-0,9994507514	-1,0000000000	-0,9994742218
-0,9396926208	0,7660444431	0,7658196037	0,7660444431	0,7658430741
-0,7660444431	-0,1736481777	-0,1736057576	-0,1736481777	-0,1736292281
-0,5000000000	-0,5000000000	-0,5000550470	-0,5000000000	-0,5000315765
-0,1736481777	0,9396926208	0,9394148618	0,9396926208	0,9393913913
0,1736481777	-0,9396926208	-0,9397003563	-0,9396926208	-0,9396768859
0,5000000000	0,5000000000	0,4999661235	0,5000000000	0,4999426530
0,7660444431	0,1736481777	0,1736939002	0,1736481777	0,1737173707
0,9396926208	-0,7660444431	-0,7658372660	-0,7660444431	-0,7658607365
1,0000000000	1,0000000000	1,0004865025	1,0000000000	1,0005099730

Tabela 2: Wyniki działania programu (c.d.)

$x$	$\tilde{I}_9(x) - I_9(x)$
-0,9876883406	0,0002770237
-0,8910065242	-0,0002969885
-0,7071067812	0,0001622344
-0,4539904997	-0,0001550733
-0,1564344650	-0,0002139034
0,1564344650	-0,0000289128
0,4539904997	-0,0000187777
0,7071067812	0,0000162085
0,8910065242	0,0001063884
0,9876883406	0,0003657385

$x$	$\tilde{J}_9(x) - J_9(x)$	$\tilde{K}_8(x) - K_8(x)$
-1,0000000000	0,0005492486	0,0005257782
-0,9396926208	-0,0002248394	-0,0002013690
-0,7660444431	0,0000424200	0,0000189496
-0,5000000000	-0,0000550470	-0,0000315765
-0,1736481777	-0,0002777590	-0,0003012294
0,1736481777	-0,0000077356	0,0000157349
0,5000000000	-0,0000338765	-0,0000573470
0,7660444431	0,0000457226	0,0000691930
0,9396926208	0,0002071771	0,0001837066
1,0000000000	0,0004865025	0,0005099730

Tabela 3: Różnice  $\tilde{s}_n - s_n$  (niezależne od  $f$ ).

- [2] Stefan Paszkowski, *Zastosowania numeryczne wielomianów i szeregów Czebyszewa*, PWN, Warszawa 1975.
- [3] Eric W. Weinstein, *Clenshaw Recurrence Formula*, za MathWorld<sup>2</sup>.

---

<sup>2</sup><http://mathworld.wolfram.com/ClenshawRecurrenceFormula.html>