

Pracownia z ANALIZY NUMERYCZNEJ

Lista nr 2

Początek zapisów: 14 listopada 2016 r.

Termin realizacji: 11 grudnia 2016 r.

Punktacja (podana przy każdym zadaniu): 8-12 punktów

Każde z zadań może być wybrane najwyżej przez trzy osoby (trzy zespoły dwuosobowe — w wypadku zadań P2.7, P2.20) spośród wszystkich zapisanych na pracownię.

P2.1. 12 punktów Rozważmy metodę iteracyjną Halleya,

$$x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k) - \frac{f''(x_k)}{2f'(x_k)}f(x_k)}, \quad (1)$$

oraz metodę iteracyjną quasi-Halleya,

$$x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k) - \frac{f'(x_k) - f'(x_{k-1})}{2(x_k - x_{k-1})f'(x_k)}f(x_k)}. \quad (2)$$

Obie metody służą do rozwiązywania równań nieliniowych.

- Pokaż w jaki sposób wykorzystać metodę Newtona do wyprowadzenia wzoru (1).
- Jaki jest rząd zbieżności metody Halleya?
- Pokaż w jaki sposób wykorzystać wzór (1) do wyprowadzenia wzoru (2).
- Jaki jest rząd zbieżności metody quasi-Halleya?

Wybierz kilka przykładów i porównaj obie metody w praktyce.

Literatura:

- [1] A. Ben-Israel, *Newton's Method with Modified Functions*, Contemporary Mathematics 204 (1997), 39–50.

P2.2. 8 punktów Rozważmy następującą modyfikację metody siecznych. Aby przybliżyć $f'(x_n)$ we wzorze związanym z metodą Newtona,

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)},$$

wykorzystujemy interpolację kwadratową w węzłach x_n, x_{n-1}, x_{n-2} . Pokaż, że w rezultacie otrzymujemy

$$x_{n+1} = x_n - \frac{f(x_n)}{\omega},$$

gdzie $\omega = f[x_n, x_{n-1}] + (x_n - x_{n-1})f[x_n, x_{n-1}, x_{n-2}]$. Wykorzystując kilka wybranych przykładów, porównaj to podejście z metodami Newtona i siecznych.

P2.3. 10 punktów Niech p_n oznacza wielomian drugiego stopnia interpolujący

$$(x_{n-2}, f(x_{n-2})), (x_{n-1}, f(x_{n-1})), (x_n, f(x_n)).$$

Następnie, zdefiniujmy x_{n+1} jako miejsce zerowe wielomianu p_n , które jest najbliższe x_n . Wykorzystaj powyższy pomysł do wyprowadzenia metody iteracyjnej Müllera rozwiązującej równania nieliniowe postaci $f(x) = 0$. Jaka jest interpretacja graficzna tej metody? Przetestuj uzyskaną metodę dla kilku wybranych nieliniowych funkcji f i porównaj ją z innymi znanymi metodami.

P2.4. 11 punktów Na podstawie zadanego ciągu par,

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

oraz danej wartości c , znajdź punkt x , dla którego $y(x) = c$. Najprostsze podejście do tego problemu polega na interpolacji punktów (y_i, x_i) . Ten sposób jest dobry tylko, gdy $x(y)$ można dobrze przybliżyć wielomianem. Załóżmy, że funkcję $y(x)$ można dobrze przybliżyć wielomianem, natomiast $x(y)$ już nie. Wykorzystując postać Newtona wielomianu interpolującego funkcję $y(x)$,

$$y_0 + y[x_0, x_1](x - x_0) + \sum_{i=2}^n y[x_0, x_1, \dots, x_i]p_i(x) = c,$$

gdzie $p_i(x) = \prod_{j=0}^{i-1} (x - x_j)$, wyprowadź metodę iteracyjną służącą do wyznaczania wartości x . Zaproponuj kilka przykładów, aby przetestować metodę.

Literatura:

- [1] G. Dahlquist, Å. Björck, *Numerical Methods in Scientific Computing*, Volume 1, SIAM, 2008.

P2.5. 10 punktów Niech dane będą wartości funkcji dwóch zmiennych,

$$f_{ij} := f(x_i, y_j) \quad (i = 1, 2, \dots, n; j = 1, 2, \dots, m),$$

gdzie x_i oraz y_j są dane. Interpolacja funkcji dwóch zmiennych polega na znalezieniu wielomianu

$$p(x, y) = \sum_{i=0}^{n-1} \sum_{j=0}^{m-1} c_{ij} x^i y^j,$$

który spełnia warunki

$$p(x_i, y_j) = f_{ij} \quad (i = 1, 2, \dots, n; j = 1, 2, \dots, m).$$

Zaproponuj algorytm rozwiązujący to zadanie. Przetestuj uzyskaną metodę dla kilku wybranych funkcji f .

Literatura:

- [1] G. Dahlquist, Å. Björck, *Numerical Methods in Scientific Computing*, Volume 1, SIAM, 2008.

P2.6. 10 punktów Zrealizować algorytmy obliczania wartości wielomianu podanego za pomocą wzoru interpolacyjnego Lagrange'a oraz zamiany postaci Lagrange'a na postać Newtona. Porównać dokładności wyników uzyskanych za pomocą obu wzorów m.in. dla funkcji $f_1(x) = (1 + 25x^2)^{-1}$ i $f_2(x) = \arctg x$.

Literatura: W. Werner, *Mathematics of Computation* 43 (1984), 205–217; Kopię artykułu można otrzymać od prowadzących pracownię.

P2.7. Zadanie dla dwuosobowego zespołu. 12 punktów Niech p_N będzie wielomianem stopnia $N > 1$ interpolującym daną funkcję f w węzłach $t_s = \cos \frac{\pi s}{N}$ ($s = 0, 1, \dots, N$). Wielomian p_N można podać wzorem

$$p_N \equiv \sum_{k=0}^N {}'' b_k^N T_k \stackrel{\text{def}}{=} \frac{b_0^N}{2} T_0 + \sum_{k=1}^{N-1} b_k^N T_k + \frac{b_N^N}{2} T_N,$$

gdzie T_k jest k -tym wielomianem Czebyszewa, a

$$b_k^N = \frac{2}{N} \sum_{s=0}^N {}'' f\left(\cos \frac{\pi s}{N}\right) \cos \frac{\pi k s}{N} \quad (k = 0, 1, \dots, N).$$

Używając algorytmu Clenshawa można obliczyć b_k^N ($k = 0, 1, \dots, N$) kosztem $O(N^2)$ działań. Wielkości te można jednak wyznaczyć szybciej. Mianowicie, dowodzi się, że

$$b_k^N = \frac{1}{N} \sum_{s=0}^{2N-1} f\left(\cos \frac{2\pi s}{2N}\right) \exp\left(\frac{2\pi i k s}{2N}\right) \quad (k = 0, 1, \dots, N).$$

Zakładając, że $N = 2^m$ ($m > 1$) i wykorzystując powyższy wzór, współczynniki b_k^N dla $k = 0, 1, \dots, N$ można obliczyć w czasie $O(N \log N)$, za pomocą tzw. *szybkiej transformacji Fouriera*. Patrz np. [1], [2] i [3]. Dla kilku odpowiednio dobranych funkcji f , m.in. e^x , $\sin(e^{2x})$, $\sqrt{1-x^2}$, $\cos(|4x|)$ oraz funkcji Rungego, porównać różne algorytmy wyznaczania wielomianu interpolacyjnego p_N dla $N = 2^m$ ($m = 2, 3, \dots, 30$), zarówno pod względem dokładności jak i stabilności oraz efektywności. Przez dokładność rozumiemy wartość błędu

$$\max_{-1 \leq x \leq 1} |f(x) - p_N(x)|,$$

a przez stabilność to, czy błąd ten maleje wraz ze wzrostem m (teoretycznie, jeśli funkcja f jest ciągła i ma wahanie ograniczone, to ciąg wielomianów p_N jest do niej zbieżny jednostajnie, gdy $N \rightarrow \infty$, na odcinku $[-1, 1]$).

Literatura

- [1] G. Dahlquist, Å. Björck, *Numerical Methods in Scientific Computing*, Volume 1, SIAM, 2008.
 [2] P. J. Davis, P. Rabinowitz, *Methods of Numerical Integration*, Second ed., Academic Press, New York, 1984.
 [3] D. Kincaid, W. Cheney, *Analiza numeryczna*, WNT, 2006.

P2.8. 9 punktów Zrealizować algorytm obliczania wartości wielomianu $H_{2k+1} \in \Pi_{2k+1}$, spełniającego dla danych parami węzłów x_0, x_1, \dots, x_k i danych liczb $y_0, y_1, \dots, y_k, y'_0, y'_1, \dots, y'_k$ warunki

$$H_{2k+1}(x_i) = y_i, \quad H'_{2k+1}(x_i) = y'_i \quad (i = 0, 1, \dots, k).$$

Rozważyć różne warianty wzoru dla wielomianu H_{2k+1} . Wykonać doświadczenia podobne do opisanych w pracy W. Werner, Mathematics of Computation 43 (1984), 205–217 (kopię artykułu można otrzymać od prowadzących pracownię).

P2.9. 8 punktów Skonstruować **naturalną funkcję sklejaną III stopnia** s , interpolującą daną funkcję f w $n+1$ parami różnych punktach przedziału $[a, b]$. Obliczyć błąd

$$E_N^{(n)} := \max_{x \in D_N} |f(x) - s(x)|,$$

gdzie D_N jest zbiorem N równoodległych punktów przedziału $[a, b]$. Wykonać obliczenia dla kilku par wartości n i N oraz dla funkcji (a) $f(x) = \sin x$, $0 \leq x \leq \pi$; (b) $f(x) = e^x$, $0 \leq x \leq 4$; (c) $f(x) = (x^2 + 1)^{-1}$, $x \in [-5, 5]$; (d) $f(x) = x/(x^2 + \frac{1}{4})$, $x \in [-\pi, \pi]$.

P2.10. 10 punktów Wartości funkcji f znane są jedynie w punktach $x_0 < x_1 < \dots < x_n$ ($n \in \mathbb{N}$). Zaproponuj, jak wykorzystać naturalną funkcję sklejaną trzeciego stopnia do znalezienia przybliżonych wartości wszystkich rozwiązań równania $f(x) = c$ ($c \in \mathbb{R}$) leżących w przedziale (x_0, x_n) . Wykonując odpowiednie testy numeryczne, zbadaj czy pomysł ten sprawdza się w praktyce.

P2.11. 10 punktów Dla danej liczby naturalnej n , danych węzłów x_0, x_1, \dots, x_n ($a = x_0 < x_1 < \dots < x_n = b$) i danej funkcji f istnieje dokładnie jedna funkcja $\tilde{s} \in C^2[a, b]$, zwana **okresową funkcją sklejaną interpolacyjną III stopnia**, spełniająca następujące warunki:

1° w każdym z podprzedziałów $[x_{k-1}, x_k]$ ($k = 1, 2, \dots, n$) funkcja \tilde{s} jest identyczna z pewnym wielomianem stopnia co najwyżej trzeciego;

2° $\tilde{s}(x_k) = f(x_k)$ ($k = 0, 1, \dots, n$; $f(x_n) = f(x_0)$);

3° $\tilde{s}^{(i)}(a+0) = \tilde{s}^{(i)}(b-0)$ ($i = 0, 1, 2$).

Dla danej zamkniętej krzywej parametrycznej $x = x(t)$, $y = y(t)$ ($a \leq t \leq b$; $x(a) = x(b)$, $y(a) = y(b)$) skonstruować następującą **zamkniętą krzywą sklejaną interpolacyjną**. Dla wybranych: $n \in \mathbb{N}$ oraz $a = t_0 < t_1 < \dots < t_n = b$ obliczamy $x_i = x(t_i)$, $y_i = y(t_i)$ dla $i = 0, 1, \dots, n$, a następnie konstruujemy okresowe funkcje sklejane interpolujące III stopnia $\tilde{s}_x(t)$, $\tilde{s}_y(t)$. Poszukiwana krzywa sklejana ma przedstawienie parametryczne $x = \tilde{s}_x(t)$, $y = \tilde{s}_y(t)$ ($a \leq t \leq b$). Wykonać obliczenia m.in. dla okręgu, elipsy i następujących danych:

(3.7, 6.4)	(3.2, 6.7)	(2.7, 6.5)	(2.1, 6.4)	(1.7, 6.0)	(1.1, 5.9)	(0.7, 5.7)	(0.4, 5.7)
(0.4, 5.4)	(0.5, 5.0)	(0.3, 4.6)	(0.6, 4.3)	(0.6, 4.0)	(0.7, 3.7)	(0.6, 3.2)	(0.8, 2.9)
(0.8, 2.6)	(0.6, 2.4)	(0.8, 2.3)	(0.9, 2.4)	(1.1, 2.2)	(1.4, 2.1)	(1.8, 2.0)	(1.7, 1.8)
(1.9, 1.4)	(2.2, 1.5)	(2.1, 1.8)	(2.7, 1.6)	(2.6, 1.4)	(3.3, 1.3)	(3.5, 0.9)	(3.7, 0.6)
(3.9, 0.8)	(4.2, 0.7)	(4.3, 0.4)	(4.5, 0.5)	(4.7, 0.7)	(5.0, 0.6)	(5.5, 0.8)	(5.9, 0.6)
(6.2, 0.4)	(6.4, 0.3)	(6.3, 0.7)	(6.5, 1.2)	(6.8, 1.7)	(7.2, 2.0)	(7.1, 2.2)	(7.2, 2.4)
(6.8, 2.8)	(6.7, 3.2)	(6.8, 3.6)	(6.4, 3.9)	(6.2, 4.2)	(6.9, 4.5)	(6.8, 5.1)	(6.6, 5.6)
(6.5, 6.0)	(6.1, 6.2)	(5.5, 6.1)	(5.0, 6.2)	(4.6, 6.2)	(4.1, 6.3)	(3.7, 6.0)	(3.4, 6.1)
(3.2, 6.5)	(3.7, 6.4)						

TABELA 1. Tajemnicza krzywa, zawarta w kwadracie $[0, 7.5] \times [0, 7.5]$

P2.12. 10 punktów Na papierze milimetrowym narysować (jednym pociągnięciem!) kontur ulubionego zwierzątka. Wybrać n (np. 10 lub 20) punktów na otrzymanej linii i ponumerować je:

$$(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n). \quad (3)$$

Odtworzyć w przybliżeniu zadaną linię, stosując jeden lub oba z następujących pomysłów:

- podzielić ciąg (3) na takie podciągi, żeby każdy z nich zawierał punkty leżące na wykresie pewnej funkcji; uzyskać przybliżoną postać linii wzorcowej łącząc wykresy przybliżeń tych funkcji;
- potraktować linię jako krzywą parametryczną $[x(t), y(t)]$, gdzie t jest parametrem przebiegającym przedział $[1, n]$, tak więc $x_i = x(i)$, $y_i = y(i)$ dla $i = 1, 2, \dots, n$; zrekonstruować funkcje $x(t)$, $y(t)$ stosując interpolację. Rozważyć wariant zadania, w którym punkty (3) są podawane w pliku tekstowym. Sprawdzić działanie programu dla danych z tabeli z zadania **P2.11**.

P2.13. 10 punktów Dla danej liczby naturalnej n , danych węzłów x_0, x_1, \dots, x_n ($a = x_0 < x_1 < \dots < x_n = b$), danej liczby rzeczywistej τ i danej funkcji f istnieje dokładnie jedna taka funkcja S_τ , zwana **hiperboliczną funkcją sklejaną interpolacyjną**, że 1° $S_\tau \in C^2[a, b]$, 2° $S_\tau(x_k) = f(x_k)$ ($k = 0, 1, \dots, n$); 3° w każdym z podprzedziałów (x_k, x_{k+1}) ($k = 0, 1, \dots, n-1$) funkcja S_τ spełnia warunek $S_\tau^{(4)}(x) - \tau^2 S_\tau''(x) = 0$; 4° $S_\tau''(a) = S_\tau''(b) = 0$. Można wykazać, że dla $x \in [x_k, x_{k+1}]$ ($0 \leq k \leq n-1$) funkcja S_τ wyraża się wzorem

$$S_\tau(x) = \{M_k \sinh[\tau(x_{k+1} - x)] + M_{k+1} \sinh[\tau(x - x_k)]\} / \sinh(\tau h_k) \\ + [f(x_k) - M_k](x_{k+1} - x)/h_k + [f(x_{k+1}) - M_{k+1}](x - x_k)/h_k,$$

gdzie $h_k := x_{k+1} - x_k$, a wartości $M_i := S_\tau''(x_i)/\tau^2$ ($i = 0, 1, \dots, n$) otrzymuje się jako rozwiązanie układu równań

$$\alpha_{i-1}M_{i-1} + (\beta_{i-1} + \beta_i)M_i + \alpha_iM_{i+1} = \gamma_i - \gamma_{i-1} \quad (1 \leq i \leq n-1; M_0 = M_n = 0),$$

gdzie z kolei

$$\alpha_i := 1/h_i - \tau / \sinh(\tau h_i), \quad \beta_i := \tau \cosh(\tau h_i) / \sinh(\tau h_i) - 1/h_i, \quad \gamma_i := f[x_i, x_{i+1}].$$

Zrealizować powyższy algorytm i sprawdzić go dla wielu wartości parametru τ (im większe τ , tym krzywa $y = S_\tau$ jest mocniej naprężona; jeśli $\tau \approx 0$, krzywa ta przypomina wykres naturalnej funkcji sklepanej interpolującej).

Literatura: D. Kincaid, W. Cheney, *Analiza numeryczna*, WNT, 2005, s. 333–335.

P2.14. 11 punktów Wielomian $I_n \in \Pi_n$ interpolujący funkcję f w węzłach

$$t_{n+1,k} = \cos \frac{2k+1}{2n+2} \pi \quad (k = 0, 1, \dots, n)$$

można zapisać w postaci

$$I_n(x) = \frac{2}{n+1} \sum'_{i=0} \left(\sum_{j=0}^n f(t_{n+1,j}) T_i(t_{n+1,j}) \right) T_i(x),$$

a wielomian J_n o własności $J_n(u_{n-1,k}) = f(u_{n-1,k})$ ($k = 0, 1, \dots, n$), gdzie $u_{n-1,k} = \cos(k\pi/n)$, ($k = 0, 1, \dots, n$), można zapisać wzorem

$$J_n(x) = \frac{2}{n} \sum''_{j=0} \left(\sum_{k=0}^n f(u_{n-1,k}) T_k(u_{n-1,j}) \right) T_j(x).$$

Wielomian $K_n \in \Pi_n$ podany wzorem

$$K_n(x) = \frac{2}{n+1} \sum'_{j=0} \left(\sum_{k=0}^{n+1} f(u_{nk}) T_k(u_{nj}) \right) T_j(x)$$

jest n -tym wielomianem optymalnym w sensie aproksymacji jednostajnej dla funkcji f na zbiorze

$$\{u_{n0}, u_{n1}, \dots, u_{n,n+1}\},$$

gdzie $u_{nk} = \cos(k\pi/(n+1))$ ($k = 0, 1, \dots, n+1$). Dla wybranych funkcji f i wartości n obliczyć (w przybliżeniu) błędy aproksymacji jednostajnej funkcji f za pomocą I_n , J_n i K_n , w przedziale $[-1, 1]$.

Uwaga. Symbol \sum' oznacza sumę, której pierwszy składnik należy podzielić przez 2, a \sum'' – sumę, której pierwszy i ostatni składnik należy podzielić przez 2.

P2.15. 10 punktów Zrealizować efektywny algorytm obliczania wartości sumy

$$s_n(x) = \sum'_{i=0}^n c_i T_i(x),$$

gdzie T_k jest k -tym wielomianem Czebyszewa. Zaburzyć wartości współczynników według wzoru $\tilde{c}_k := c_k(1 + \epsilon_k)$, wybierając losowo liczby ϵ_k z małego przedziału $[-\delta, \delta]$. Porównać wartości wielomianu $s_n(x)$ dla dokładnych i dla zaburzonych współczynników. Wykonać obliczenia **między innymi** dla wielomianów I_n i J_n z zadania **P2.14**.

P2.16. 10 punktów Zrealizować algorytm konstrukcji takiego wielomianu w możliwie najniższego stopnia, który – dla danej liczby dodatniej ε – spełnia nierówność

$$\int_0^1 x[f(x) - w(x)]^2 dx < \varepsilon.$$

Wykonać obliczenia kontrolne **między innymi** dla funkcji $f(x) = \sqrt{x}$.

- P2.17.** 12 punktów Niech dana będzie funkcja $f \in C[-1, 1]$ mająca zera w punktach -1 i $+1$, tzn. $f(-1) = f(+1) = 0$. Zaproponuj i zrealizuj algorytm wyznaczania wielomianu $w_n^* \in \tilde{\Pi}_n$ ($n \geq 2$) spełniającego warunek

$$\|f - w_n^*\|_2 = \min_{w_n \in \tilde{\Pi}_n} \|f - w_n\|_2,$$

gdzie $\tilde{\Pi}_n := \{w \in \Pi_n : w(-1) = w(+1) = 0\}$, a $\|g\|_2^2 := \int_{-1}^1 [g(x)]^2 dx$. Przedstaw wnioski z eksperymentów numerycznych przeprowadzonych dla odpowiednio dobranych funkcji f .

- P2.18.** 9 punktów Zrealizować algorytm, który dla danej funkcji f ciągłej w przedziale $[a, b]$, liczby naturalnej n oraz układu $n+2$ punktów $D_n := \{x_0, x_1, \dots, x_{n+1}\} \subset [a, b]$ wyznacza n -ty wielomian optymalny w_n^* w sensie aproksymacji jednostajnej dla funkcji f na zbiorze D_n . Wykonać obliczenia dla wybranych funkcji f i wartości n w wypadku, gdy

$$(i) \ x_k = a + k \frac{b-a}{n+1}; \quad (ii) \ x_k = \frac{a+b}{2} - \frac{b-a}{2} \cos \frac{k\pi}{n+1} \quad (k = 0, 1, \dots, n+1).$$

Naszkicować wykres funkcji $e_n := f - w_n^*$ w przedziale $[a, b]$.

- P2.19.** 12 punktów Opracuj algorytm przekształcania postaci

$$f_N(x) := \sum_{n=0}^N A(n)p_n(x),$$

do innej postaci

$$f_N(x) := \sum_{n=0}^N A^*(n)q_n(x),$$

gdzie ciągi wielomianów $\{p_n\}$, $\{q_n\}$ spełniają następujące zależności rekurencyjne:

$$p_{n+1}(x) + (a(n) + b(n)x)p_n(x) + c(n)p_{n-1}(x) = 0, \\ p_{-1}(x) = 0, \quad p_0(x) = 1, \quad 0 \leq n \leq N-1,$$

$$q_{n+1}(x) + (a^*(n) + b^*(n)x)q_n(x) + c^*(n)q_{n-1}(x) = 0, \\ q_{-1}(x) = 0, \quad q_0(x) = 1, \quad 0 \leq n \leq N-1.$$

Przetestuj algorytm na wybranych przykładach, m.in. zamień postać potęgową wielomianu na kombinację liniową wielomianów Czebyszewa.

Literatura

- [1] J. Wimp, *Computation with Recurrence Relations*, Pitman Advanced Publishing Program, Boston, 1984.

- P2.20.** Zadanie dla dwuosobowego zespołu. 12 punktów Niech b_0, b_1, \dots, b_n będzie bazą przestrzeni P_n z iloczynem skalarnym $\langle \cdot, \cdot \rangle : P_n \times P_n \rightarrow \mathbb{R}$. Funkcje $d_0^{(n)}, d_1^{(n)}, \dots, d_n^{(n)}$ stanowią bazę dualną przestrzeni P_n względem iloczynu skalarnego $\langle \cdot, \cdot \rangle$, jeżeli spełniają następujące warunki:

$$\begin{cases} \text{span} \{d_0^{(n)}, d_1^{(n)}, \dots, d_n^{(n)}\} = P_n, \\ \langle b_i, d_j^{(n)} \rangle = \delta_{ij} \quad (0 \leq i, j \leq n), \end{cases} \quad (4)$$

gdzie δ_{ij} wynosi 1, jeżeli $i = j$ oraz 0 w przeciwnym wypadku. Dla danej bazy b_0, b_1, \dots, b_n , należy opracować algorytm wyznaczania bazy dualnej $d_0^{(n)}, d_1^{(n)}, \dots, d_n^{(n)}$, spełniającej warunki (4). Następnie, pokaż w jaki sposób wykorzystać własności baz dualnych do znalezienia elementu optymalnego p^* przestrzeni P_n , dla funkcji f , w sensie normy średniokwadratowej,

$$\|f - p^*\| = \min_{p \in P_n} \|f - p\|,$$

gdzie $\|\cdot\| = \sqrt{\langle \cdot, \cdot \rangle}$. W sprawozdaniu należy podać przykłady zastosowania opracowanej metody.

Literatura:

- [1] P. Woźny, Construction of dual bases, *Journal of Computational and Applied Mathematics* 245 (2013), 75–85.
 [2] P. Woźny, Construction of dual B-spline functions, *Journal of Computational and Applied Mathematics* 260 (2014), 301–311.
 [3] P. Gospodarczyk, P. Woźny, A new property of dual bases and its application, <https://arxiv.org/abs/1511.08264>.