

Wykład 4.

Algorytmy numeryczne

Algorytm Mallata. Przypomnijmy, że dla danej analizy MRA i dowolnego $j \in \mathbf{Z}$ przestrzeń V_j rozkłada się na dwie wzajemnie prostopadłe podprzestrzenie

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

Zauważmy, że zgodnie z tym, co wiemy o analizie wielorozdzielczej, w przestrzeni V_j mamy bazę ortonormalną

$$(1) \quad \{\varphi_{j,n}(x); n \in \mathbf{Z}\},$$

w przestrzeni V_{j-1} bazę o. n.

$$\{\varphi_{j-1,n}(x); n \in \mathbf{Z}\},$$

oraz w przestrzeni W_{j-1} bazę o. n.

$$\{\psi_{j-1,n}(x); n \in \mathbf{Z}\},$$

gdzie φ jest funkcją skalującą analizy, ψ związaną z nią falką, oraz używamy następujących oznaczeń

$$\varphi_{j,n}(x) = 2^{j/2} \varphi(2^j x - n), \quad \psi_{j,n}(x) = 2^{j/2} \psi(2^j x - n).$$

W takim razie w przestrzeni V_j mamy do dyspozycji dwie różne bazy o. n., bazę (1), oraz bazę

$$(2) \quad \{\varphi_{j-1,n}(x), \psi_{j-1,n}(x); n \in \mathbf{Z}\}.$$

Ważną własnością falek jest to, że istnieje szybki algorytm pozwalający obliczać współczynniki bazowe danej funkcji $f \in V_j$ w jednej z powyższych baz przy pomocy współczynników bazowych w drugiej. Jest to tak zwany algorytm Mallata, będący podstawą dyskretniej transformaty falkowej (często nazywanej w skrócie DWT).

Niech $f \in L^2(\mathbf{R})$, i wprowadźmy następujące oznaczenia na ciągi współczynników bazowych f w różnych bazach:

$$(3) \quad a_j(n) = \langle f, \varphi_{j,n} \rangle, \quad d_j(n) = \langle f, \psi_{j,n} \rangle.$$

Przypomnijmy, że ciąg

$$\{h_n\}_{n=-\infty}^{\infty}$$

oznacza filtr dolnoprzepustowy analizy, zdefiniowany przez

$$\frac{1}{2} \varphi\left(\frac{x}{2} - x\right) = \sum_{n=-\infty}^{\infty} h_n \varphi(x - n),$$

a $\{g_n\}$ oznacza filtr górnoprzepustowy, dany przez

$$g_n = (-1)^{n-1} \overline{h_{1-n}},$$

przy czym falka ψ dana jest przez

$$\frac{1}{2} \psi\left(\frac{x}{2} - x\right) = \sum_{n=-\infty}^{\infty} g_n \varphi(x - n).$$

Następujące twierdzenie przedstawia algorytm.

Twierdzenie Mallata. *Transformata:*

$$(4) \quad \begin{aligned} a_{j-1}(k) &= \sum_{n=-\infty}^{\infty} \sqrt{2h_n - 2k} a_j(n), \\ d_{j-1}(k) &= \sum_{n=-\infty}^{\infty} \sqrt{2g_n - 2k} a_j(n). \end{aligned}$$

Transformata odwrotna:

$$(5) \quad a_j(n) = \sum_{k=-\infty}^{\infty} \sqrt{2h_{n-2k}} a_{j-1}(k) + \sum_{k=-\infty}^{\infty} \sqrt{2g_{n-2k}} a_{j-1}(k).$$

Uwaga. Zauważmy, że algorytm odwołuje się tylko do filtrów dolno- i górnoprzepustowego. Same wartości funkcji φ i ψ nigdzie nie są potrzebne. Zauważmy też, że wzory powyższe są tym efektywniejsze numerycznie, im krótsze są filtry, a jeżeli są nieskończone to im szybciej maleją.

dowód tw. Mallata. Przypomnijmy wzory

$$(6) \quad \begin{aligned} h_n &= \left\langle \frac{1}{2} \varphi\left(\frac{\cdot}{2}\right), \varphi(\cdot - n) \right\rangle, \\ g_n &= \left\langle \frac{1}{2} \psi\left(\frac{\cdot}{2}\right), \varphi(\cdot - n) \right\rangle. \end{aligned}$$

Rozważmy wzór na h_n . n jest dowolne, więc wpiszmy w jego miejsce $n - 2l$, gdzie $n, l \in \mathbf{Z}$.

$$\begin{aligned} h_{n-2l} &= \left\langle \frac{1}{2} \varphi\left(\frac{\cdot}{2}\right), \varphi(\cdot - (n - 2l)) \right\rangle \\ &= \int_{-\infty}^{\infty} \frac{1}{2} \varphi\left(\frac{x}{2}\right) \overline{\varphi(x - n + 2l)} dx \\ &= \int_{-\infty}^{\infty} \frac{1}{2} \varphi\left(\frac{x - 2l}{2}\right) \overline{\varphi(x - n)} dx \\ &= \int_{-\infty}^{\infty} \frac{1}{2} \varphi\left(\frac{x}{2} - l\right) \overline{\varphi(x - n)} dx. \end{aligned}$$

Dokonujemy zamiany zmiennych $x \mapsto 2^j x$. Granice całkowania nie zmieniają się

$$\begin{aligned} h_{n-2l} &= \int_{-\infty}^{\infty} \frac{1}{2} \varphi(2^{j-1}x - l) \overline{\varphi(2^j - n)} 2^j dx \\ &= \int_{-\infty}^{\infty} \frac{1}{\sqrt{2}} \varphi_{j-1,l}(x) \overline{\varphi_{j,n}(x)} dx \\ &= \frac{1}{\sqrt{2}} \langle \varphi_{j-1,n}, \varphi_{j,n} \rangle. \end{aligned}$$

Podobnie wyprowadzamy wzór dla g_{n-2l} i otrzymujemy

$$(7) \quad \begin{aligned} \sqrt{2} h_{n-2l} &= \langle \varphi_{j-1,n}, \varphi_{j,n} \rangle, \\ \sqrt{2} g_{n-2l} &= \langle \psi_{j-1,n}, \varphi_{j,n} \rangle. \end{aligned}$$

Obliczyliśmy współczynniki bazowe funkcji $\varphi_{j-1,l}$ i $\psi_{j-1,l}$ w bazie (1), więc funkcje te mają rozwinięcia

$$(8) \quad \begin{aligned} \varphi_{j-1,l} &= \sum_{n=-\infty}^{\infty} \sqrt{2} h_{n-2l} \varphi_{j,n}, \\ \psi_{j-1,l} &= \sum_{n=-\infty}^{\infty} \sqrt{2} h_{n-2l} \varphi_{j,n}. \end{aligned}$$

Wzory te wstawiamy do definicji naszych ciągów

$$\begin{aligned} a_{j-1}(k) &= \langle f, \varphi_{j-1,k} \rangle \\ &= \left\langle f, \sum_{n=-\infty}^{\infty} \sqrt{2} h_{n-2l} \varphi_{j,n} \right\rangle \\ &= \sum_{n=-\infty}^{\infty} \sqrt{2} \overline{h_{n-2l}} \langle f, \varphi_{j,n} \rangle \\ &= \sum_{n=-\infty}^{\infty} \sqrt{2} \overline{h_{n-2l}} a_j(n). \end{aligned}$$

Podobnie obliczamy $d_{j-1}(k)$. Przypomnijmy, że rozwinięcia (8) są zbieżne w $L^2(\mathbf{R})$, więc iloczyny skalarne możemy liczyć wyraz za wyrazem. W ten sposób pokazaliśmy pierwszą część twierdzenia.

Wzory (7) możemy też potraktować jako wzory na współczynniki bazowe funkcji $\varphi_{j,n} \in V_j$ w bazie (2). W takim razie

$$(9) \quad \begin{aligned} \varphi_{j,n} &= \sum_{k=-\infty}^{\infty} \langle \varphi_{j,n}, \varphi_{j-1,k} \rangle \varphi_{j-1,k} + \sum_{k=-\infty}^{\infty} \langle \varphi_{j,n}, \psi_{j-1,k} \rangle \psi_{j-1,k} \\ &= \sum_{k=-\infty}^{\infty} \sqrt{2} \overline{h_{n-2l}} \varphi_{j-1,k} + \sum_{k=-\infty}^{\infty} \sqrt{2} \overline{g_{n-2l}} \psi_{j-1,k}. \end{aligned}$$

Możemy więc obliczyć współczynniki bazowe f

$$\begin{aligned} a_j(n) &= \langle f, \varphi_{j,n} \rangle \\ &= \left\langle f, \sum_{k=-\infty}^{\infty} \sqrt{2} \overline{h_{n-2l}} \varphi_{j-1,k} + \sum_{k=-\infty}^{\infty} \sqrt{2} \overline{g_{n-2l}} \psi_{j-1,k} \right\rangle \\ &= \sum_{k=-\infty}^{\infty} \sqrt{2} h_{n-2l} \langle f, \varphi_{j-1,k} \rangle + \sum_{k=-\infty}^{\infty} \sqrt{2} g_{n-2l} \langle f, \psi_{j-1,k} \rangle \\ &= \sum_{k=-\infty}^{\infty} \sqrt{2} h_{n-2k} a_{j-1}(k) + \sum_{k=-\infty}^{\infty} \sqrt{2} g_{n-2k} d_{j-1}(k), \end{aligned}$$

co kończy dowód. $\square$

Przyjrzyjmy się wzorom. Niech ciągi $\{\tilde{h}_n\}$ i $\{\tilde{g}_n\}$ mają współczynniki

$$\tilde{h}_n = \overline{h_{-n}}, \quad \tilde{g}_n = \overline{g_{-n}}.$$

Wtedy, zgodnie z twierdzeniem Mallata

$$a_{j-1}(k) = \sum_{n=-\infty}^{\infty} \sqrt{2} \tilde{h}_{2k-n} a_j(n) = \sqrt{2} (\tilde{h} * a_j)(2k),$$

$$d_{j-1}(k) = \sum_{n=-\infty}^{\infty} \sqrt{2} \tilde{g}_{2k-n} a_j(n) = \sqrt{2} (\tilde{g} * a_j)(2k).$$

Obliczenie ciągów a_{j-1} i d_{j-1} sprowadza się więc do splotu wyjściowego ciągu a_j z ciągiem $\tilde{h}$ i $\tilde{g}$, a następnie wybraniu co drugiego elementu powstałych ciągów. Po angielsku nazywa się to *downsampling*. W teorii przetwarzania sygnału operacje na sygnałach często opisuje się przy pomocy schematów blokowych. Naszą operację możemy przedstawić schematycznie następująco

Jeden krok dyskretnej transformaty falkowej

Podobnie przyjrzyjmy się drugiej części twierdzenia Mallata. Wprowadźmy następujące oznaczenie. Jeżeli dany jest ciąg $\{\alpha_n\}$, to przez $\{\alpha'_n\}$ oznaczamy jego „rozrzedzenie”:

$$\alpha'_n = \begin{cases} \alpha_{n/2} & : n - \text{parzyste} \\ 0 & : n - \text{nieparzyste.} \end{cases}$$

Ciąg $\{\alpha_n\}$ został więc rozrzedzony zerami:

$$\dots, \alpha_{-1}, \alpha_0, \alpha_1, \dots \quad \Rightarrow \quad \dots, \alpha_{-1}, 0, \alpha_0, 0, \alpha_1, 0, \alpha_2, \dots$$

Wzór (5) można wtedy zapisać

$$\begin{aligned} a_j(n) &= \sum_{k=-\infty}^{\infty} \sqrt{2} h_{n-2k} a_{j-1}(k) + \sum_{k=-\infty}^{\infty} \sqrt{2} g_{n-2k} d_{j-1}(k) \\ &= \sum_{\substack{k=-\infty \\ k - \text{parzyste}}}^{\infty} \sqrt{2} h_{n-k} a_{j-1}(k/2) + \sum_{\substack{k=-\infty \\ k - \text{parzyste}}}^{\infty} \sqrt{2} g_{n-k} d_{j-1}(k/2) \\ &= \sum_{k=-\infty}^{\infty} \sqrt{2} h_{n-k} a'_{j-1}(k) + \sum_{k=-\infty}^{\infty} \sqrt{2} g_{n-k} d'_{j-1}(k) \\ &= \sqrt{2} (h * a'_{j-1})(n) + \sqrt{2} (g * d'_{j-1})(n). \end{aligned}$$

Jeden krok odwrotnej dyskretnej transformaty falkowej

Zauważmy więc, że 1 krok odwrotnej transformaty falkowej sprowadza się więc najpierw do rozrzedzenia sygnałów wejściowych a_{j-1} i d_{j-1} (po angielsku *upsampling*), a następnie splotu każdego sygnału z odpowiednim ciągiem, i w końcu złożenia (sumy). Sploty są w zasadzie z tymi samymi ciągami co w transformacie. Odwrotną transformatę możemy przedstawić w postaci schematu blokowego

Sygnały dyskretne. Sygnały dyskretne to ciągi wartości $\{f_n\}$, skończone lub nie. Na razie rozważamy sygnały 1-wymiarowe. Najczęściej taki ciąg wartości to ciąg próbek jakiegoś sygnału ciągłego, na przykład dźwięku. Analiza falkowa takiego sygnału polega wyborze analizy wielorozdzielczej (czyli wyborze falki), następnie zanurzeniu sygnału w przestrzeni V_0 , a następnie obliczeniu współczynników bazowych w bazie falkowej (ciągów $\{d_j\}$). Numerycznie cały ten proces sprowadza się do prostego i szybkiego algorytmu rekurencyjnego, w którym wartości funkcji skalującej ani falki w ogóle się nie pojawiają. Jedynym elementem analizy wielorozdzielczej występującym w obliczeniach są filtry dolno- i górnoprzepustowy. Algorytm opiera się na twierdzeniu Mallata i nazywa się algorytmem Mallata. Niech naszym sygnałem będzie ciąg $\{f_n\}$. Wybrawszy analizę wielorozdzielczą przyporządkujemy sygnałowi funkcję $f \in V_0$

$$(10) \quad f(x) = \sum_{n=-\infty}^{\infty} f_n \varphi(x - n).$$

Takie przyporządkowanie jest bardzo naturalne. Najczęściej wartości f_n są próbkami jakiejś ciągłej wartości fizycznej, pobieranymi w regularnych odstępach czasu. Czujnik pobierający próbkę z reguły pobierając ją dokonuje uśrednienia wartości w jakimś przedziale czasu. Cały proces pobrania próbek i przyporządkowania im elementu $f \in V_0$ jest więc rzutem wyjściowego sygnału ciągłego na V_0 . W sumie bardzo naturalna i łagodna operacja, jeżeli analizę wielorozdzielczą dobierzemy właściwie do charakteru badanego sygnału.

Dyskretna transformata falkowa dyskretnego sygnału $\{f_n\}$ to zbiór współczynników bazowych

$$(11) \quad \{d_j(n) = \langle f, \psi_{j,n} \rangle; j, n \in \mathbf{Z}, j \leq -1\}.$$

Widzimy więc, że obliczenie transformaty sprowadza się do rekurencyjnego stosowania twierdzenia Mallata, z $a_0 = \{f_n\}$. Sygnał można odtworzyć z jego transformaty falkowej (11) stosując rekurencyjnie drugą część twierdzenia Mallata. W praktyce sygnał analizowany jest zawsze skończony i algorytm Mallata ma skończoną liczbę kroków.

Sygnały skończone. W przypadku sygnału skończonego składającego się z N próbek

$$\{f_0, f_1, \dots, f_{N-1}\}$$

algorytm wygląda następująco. Algorytm ma zastosowanie do sygnałów, których długość jest potęgą 2, więc najpierw ewentualnie wydłużamy sygnał, poprzez dodanie zer, tak, aby jego długość była potęgą 2, $N = 2^J$. Następnie określamy wyjściowy ciąg a_0 , jako N -periodyzację sygnału:

$$a_0(n) = f_n, \quad n = 0, \dots, N-1,$$

oraz

$$a_0(n + kN) = a_0(n), \quad k \in \mathbf{Z}, \quad n = 0, \dots, N-1.$$

Łatwo zauważyć, że zastosowanie 1 kroku transformaty falkowej do nam 2 ciągi

$$\{a_{-1}(n)\}_{n=-\infty}^{\infty} \quad \text{i} \quad \{d_{-1}(n)\}_{n=-\infty}^{\infty},$$

które są okresowe o okresach dwukrotnie krótszych

$$a_{-1}(n + N/2) = a_{-1}(n), \quad d_{-1}(n + N/2) = d_{-1}(n).$$

Wynika to wprost ze wzorów, na przykład

$$\begin{aligned} a_{-1}(n + N/2) &= \sum_{k=-\infty}^{\infty} \sqrt{2} \overline{h_{k-2(n+N/2)}} a_0(k) \\ &= \sum_{k=-\infty}^{\infty} \sqrt{2} \overline{h_{k-2n-N}} a_0(k) \\ &= \sum_{k=-\infty}^{\infty} \sqrt{2} \overline{h_{k-2n}} a_0(k + N) \\ &= \sum_{k=-\infty}^{\infty} \sqrt{2} \overline{h_{k-2n}} a_0(k) \\ &= a_{-1}(n). \end{aligned}$$

Ze względu na tę okresowość sygnały a_{-1} i d_{-1} są o połowę krótsze, bo wystarczy zapamiętywać wartości odpowiadające długości okresu. Iterując procedurę, po J krokach uzyskujemy kompletną transformatę falkową wyjściowego sygnału $\{f_0, \dots, f_{N-1}\}$:

$$a_{-J}(0), d_{-J}(0), d_{-J+1}(0), d_{-J+1}(1), \dots, d_{-1}(0), \dots, d_{-1}(N/2 - 1).$$

Odwrotną transformatę obliczamy podobnie. Każdy ze skończonych ciągów d_j przedłużamy okresowo do ciągu nieskończonego, długość okresu taka jak długość ciągu, następnie stosujemy J razy twierdzenie Mallata. W każdym kroku rekonstruujemy ciąg a_j , o coraz dłuższym okresie.