

Filtry – właściwości, projektowanie, przetwarzanie sygnałów

4.1. Podstawy teoretyczne

Najczęściej spotykane filtry cyfrowe są to układy liniowe i stacjonarne – a więc składają się z operacji liniowych wykonywanych na przetwarzanym sygnale: opóźnienia, mnożenia przez stałą, sumowania.

W uzasadnionych przypadkach stosuje się inne filtry – nieliniowe albo nie-stacjonarne, ale ich analiza jest znacznie trudniejsza. W opisie filtrów liniowych i stacjonarnych można stosować takie bardzo przydatne pojęcia, jak:

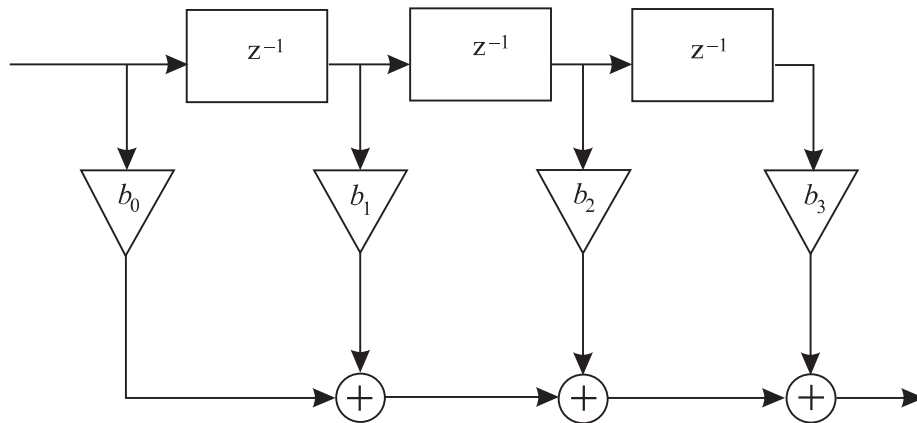
- odpowiedź impulsowa $h[n]$,
- transmitancja $H(z)$ albo $H(e^{j\theta})$,
- charakterystyki częstotliwościowe¹:
 - amplitudowa $A(\theta) = |H(e^{j\theta})|$,
 - fazowa $\varphi(\theta) = \arg H(e^{j\theta})$,
- opóźnienie grupowe $\tau(\theta) = -\frac{d\varphi(\theta)}{d\theta}$.

Należy zauważyć, że fizyczne realizacje filtrów cyfrowych są liniowe tylko w przybliżeniu – pracują na sygnałach skwantowanych, mnożenie wykonują ze skończoną precyzją, i mają ograniczony zakres przetwarzanych liczb.

4.1.1. Uśrednianie z wagami, czyli filtry SOI (ang. *FIR*)

Najprostsze w realizacji i w analizie oraz projektowaniu są filtry, w których pewna liczba przeszłych próbek jest zapamiętywana, a następnie sumowana z wagami (rys. 4.1). Zauważ, że na schematach filtrów z^{-1} oznacza opóźnienie sygnału o jedną próbkę.

¹ Po angielsku najczęściej używa się terminu „frequency response”, a więc „odpowiedź częstotliwościowa”; po polsku zamiast *amplitudowa charakterystyka częstotliwościowa* często pisze się skrótowo „charakterystyka amplitudowa” albo „charakterystyka częstotliwościowa”. Oczywiście najbardziej jest używać terminu dokładnego.



Rysunek 4.1. Filtr SOI w strukturze transwersalnej

Jak łatwo sobie wyobrazić, patrząc na rysunek, filtr taki realizuje bezpośrednio splot sygnału wejściowego z ciągiem $b[k] = \{b_0, b_1, b_2, b_3\}$.

$$y(n) = \sum_{k=0}^K x(n-k)b(k). \quad (4.1)$$

To oznacza, że ciąg współczynników $b[k]$ jest po prostu odpowiedzią impulsową tego filtru, i jest to filtr o skończonej odpowiedzi impulsowej – SOI (ang. *Finite Impulse Response – FIR*). W tym wypadku długość odpowiedzi wynosi 4.

Gdy przypomnimy sobie, że splot w dziedzinie czasu odpowiada mnożeniu w dziedzinie widma², zobaczymy natychmiast, że charakterystyka częstotliwościowa filtru to transformata Fouriera jego odpowiedzi impulsowej – a w przypadku filtru SOI ciągu jego współczynników.

Spróbujmy zapisać równanie (4.1) w dziedzinie przekształcenia \mathcal{Z} . Przekształcenie \mathcal{Z} jest liniowe [4] – to znaczy, że sumowanie i skalowanie (sygnałów) w dziedzinie n można przenieść na takie same operacje (na transformatach tych sygnałów) w dziedzinie z . Natomiast opóźnieniu w dziedzinie czasu $x(n - n_0)$ odpowiada mnożenie $X(z)$ przez z^{-n_0} .

Otrzymujemy więc

$$Y(z) = \sum_{k=0}^K X(z) \cdot z^{-k} b(k). \quad (4.2)$$

Ciągu współczynników b_k na razie nie przekształcamy – z punktu widzenia sygnału to są stałe skalujące! Możemy jednak wyciągnąć $X(z)$ przed sumę (to jest

² Uwaga, to jest proste dla sygnałów o ograniczonej energii (z widmem ciągłym – i o takich sygnałach tu mówimy) – jeśli widmo jest dyskretne, splot w dziedzinie czasu musi być splotem kołowym (okresowym).

bardzo przydatna własność przekształcenia \mathcal{Z} – możliwość rozdzielenia operatora opóźnienia od transformaty sygnału)

$$Y(z) = X(z) \sum_{k=0}^K z^{-k} b(k) = X(z) \cdot H(z). \quad (4.3)$$

W ten sposób zobaczyliśmy, że $H(z) = \sum_{k=0}^K b(k)z^{-k}$ – czyli $H(z)$ jest to transformata \mathcal{Z} odpowiedzi impulsowej filtru (a dla filtru SOI – ciągu jego współczynników).

Gdyby więc $b[n] = 1, -3, 3, -1$, to $H(z) = 1 - 3z^{-1} + 3z^{-2} - z^{-3}$. Jest to wielomian rzędu 3 (liczy się najwyższa potęga z^{-1}), czyli mamy do czynienia z filtrem rzędu 3. Zauważ, że filtr SOI rzędu 3 ma 3 bloki opóźnienia, ale 4 współczynniki.

Równie szybko uzyskamy charakterystykę częstotliwościową. Podstawiając $z = e^{j\theta}$ do wzoru definiującego przekształcenie \mathcal{Z} otrzymuje się przekształcenie Fouriera (inaczej można to sformułować „obliczając transformatę \mathcal{Z} na okręgu jednostkowym $z = e^{j\theta}$...”).

Zatem, aby otrzymać charakterystykę częstotliwościową filtru SOI, wystarczy:

- zapisać transformatę \mathcal{Z} odpowiedzi impulsowej,
- podstawić $z = e^{j\theta}$,
- i gotowe – teraz można np. obliczyć i wykreślić moduł $A(\theta) = |H(e^{j\theta})|$.

Wielomian $H(z)$ w przypadku filtru SOI określa jednoznacznie charakterystykę filtru. Ponieważ wielomian można opisać (z dokładnością do stałej skalującej) położeniem jego pierwiastków,

$$H(z) = C \cdot (z - z_1)(z - z_2)(z - z_3) \dots \quad (4.4)$$

warto się zastanowić, o czym mówią nam pierwiastki $H(z)$. Otóż położenia pierwiastków (zer transmitancji) w pobliżu okręgu jednostkowego (lub na nim) wyznaczają częstotliwości, dla których filtr ma znaczne tłumienie. Aby to zrozumieć, zapiszmy w podobnej postaci $A(\theta)$

$$A(\theta) = |H(z)|_{z=e^{j\theta}} = |C| \cdot |e^{j\theta} - z_1| \cdot |e^{j\theta} - z_2| \cdot |e^{j\theta} - z_3| \cdot \dots \quad (4.5)$$

Widać, że zawsze, gdy odległość kartezjańska punktu $z = e^{j\theta}$ od jakiegoś zera z_i jest niewielka, $A(\theta)$ jest pod silnym wpływem tego zera.

4.1.2. Sprzężenie zwrotne, czyli filtry NOI (ang. *IIR*)

Układy filtrów SOI zawierały operację opóźnienia tylko dla sygnału wejściowego. Jeśli do sumy wprowadzimy także opóźniony sygnał z wyjścia, otrzymamy układ ze sprzężeniem zwrotnym. W takim układzie raz wprowadzone pobudzenie może nigdy nie zaniknąć. Dlatego tę klasę filtrów nazywamy filtrami NOI (o nieskończonej odpowiedzi impulsowej; ang. *Infinite Impulse Response* – *IIR*).

Przeanalizujemy co dzieje się w układzie jak na rys. 4.2. Otóż

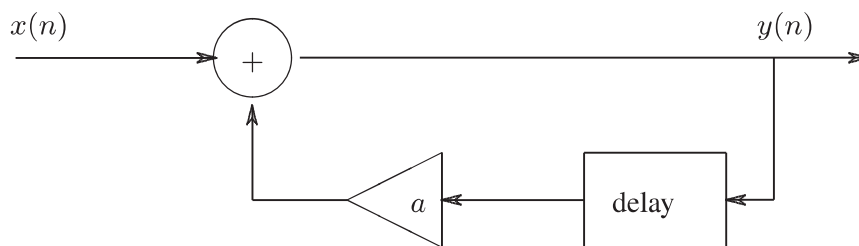
$$y(n) = x(n) + ay(n-1) \quad (4.6)$$

$$Y(z) = X(z) + aY(z)z^{-1} \quad (4.7)$$

$$Y(z) = X(z) \frac{1}{1 - az^{-1}} \quad (4.8)$$

$$H(z) = \frac{Y(z)}{X(z)} = \frac{1}{1 - az^{-1}} = \frac{z}{z - a}, \quad (4.9)$$

a więc $h(n) = a^n u(n)$ jeśli szukamy rozwiązania przyczynowego.



Rysunek 4.2. Bardzo prosty filtr NOI

Spróbujmy wyznaczyć charakterystykę amplitudową filtra:

$$H(z) = \frac{z}{z - a} \quad (4.10)$$

$$H(e^{j\theta}) = \frac{e^{j\theta}}{e^{j\theta} - a} \quad (4.11)$$

$$|H(e^{j\theta})| = \frac{1}{|e^{j\theta} - a|} \quad (4.12)$$

$$|H(e^{j\theta})| = \frac{1}{|\cos \theta - a + j \sin \theta|} = \frac{1}{\sqrt{1 - 2a \cdot \cos \theta + a^2}}. \quad (4.13)$$

Funkcję tę wykreślono na rys. 4.3 w funkcji częstotliwości unormowanej (czyli $\frac{\theta}{2\pi}$).

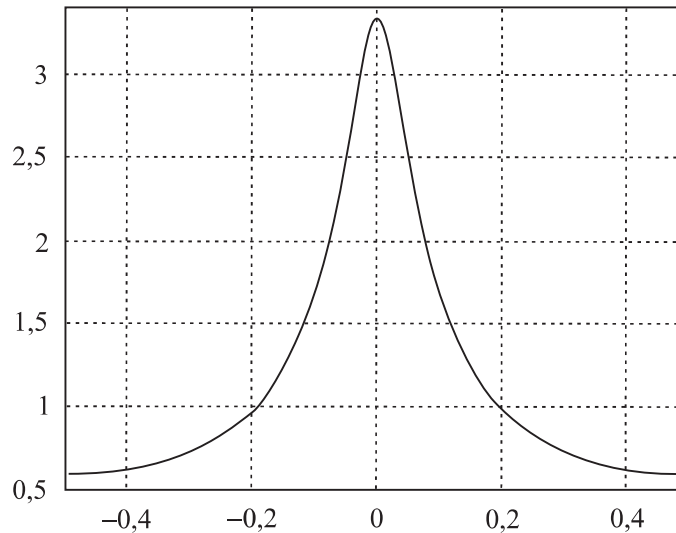
Jak widać, $H(z)$ jest funkcją wymierną, czyli ilorazem dwóch wielomianów. Wielomian mianownika ma pierwiastek w punkcie a – jest to biegun funkcji $H(z)$, i ten biegun jest odpowiedzialny za składnik $a^n u(n)$ w odpowiedzi impulsowej.

Każdą funkcję wymierną można rozłożyć na ułamki proste; w przypadku transmitancji taki rozkład pozwala łatwo znaleźć jej odwrotną transformatę \mathcal{Z} (czyli odpowiedź impulsową układu).

Jeśli $X(z)$ jest funkcją wymierną z M zerami i N różnymi³ biegunami d_k ,

$$X(z) = \sum_{r=0}^{M-N} B_r \cdot z^{-r} + \sum_{k=1}^N \frac{A_k}{1 - d_k z^{-1}}, \quad A_k = (1 - d_k z^{-1}) \cdot X(z) \Big|_{z=d_k}$$

³ Co dzieje się przy biegunach wielokrotnych, zalecamy sprawdzić w podręczniku analizy z pierwszego roku studiów.



Rysunek 4.3. Charakterystyka amplitudowa filtru z rys. 4.2

Szukanie transformaty odwrotnej z tak rozłożonego wielomianu jest proste – każdy biegun d_k generuje nam składową wykładniczą $u(n)d_k^n$. Łatwo teraz wyjaśnić, dlaczego bieguny muszą być wewnątrz okręgu jednostkowego – dla $|d_k| > 1$ taka składowa narasta wykładniczo.

Warto zauważyć, że para biegunów sprzężonych (w zapisie biegunowym: $re^{+j\theta}$ i $re^{-j\theta}$) wygeneruje razem $u(n) \cdot r^n \cdot (e^{+j\theta n} + e^{-j\theta n})$, czyli zanikającą⁴ wykładniczo kosinusoidę.

4.1.3. Projektowanie filtrów

Co to znaczy zaprojektować filtr?

- Zdefiniować wymagania (co właściwie chcemy od filtru).
- Określić, czy filtr będzie NOI czy SOI (z tego wynika inny sposób projektowania, inne niebezpieczeństwa i inne implementacje).
- Określić rząd filtru (szukamy najniższego rzędu, który pozwala spełnić wymagania – ale często zdarza się, że mamy też ograniczenie maksymalnego rzędu wynikające z możliwości realizacyjnych, i wtedy wymagania trzeba nagiąć).
- Znaleźć współczynniki filtru.
- Zdecydować o szczegółach implementacji (nie zawsze można filtr wysokiego rzędu zrealizować wprost ze wzoru; trzeba zabezpieczyć się przed kumulacją błędów numerycznych, przed przekroczeniem zakresu dynamiki itd.).

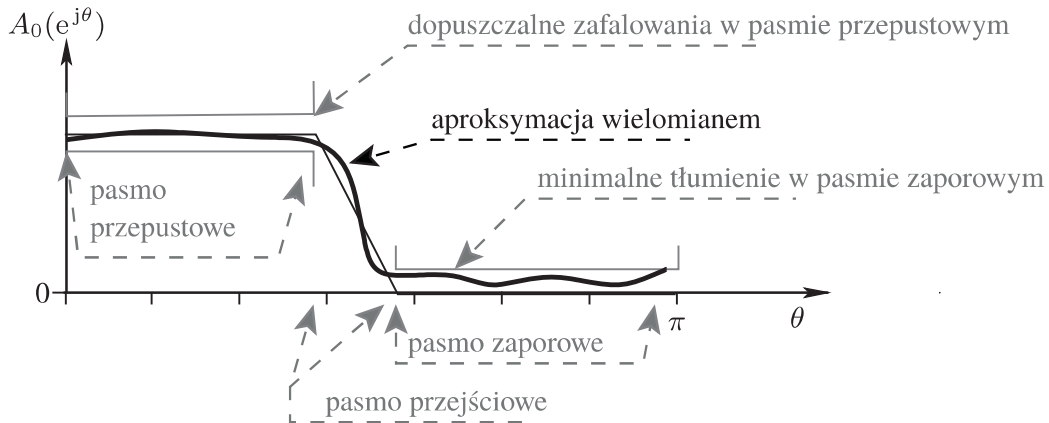
Jak opisać wymagania na charakterystykę filtru?

- Zdecydować, dla jakich częstotliwości filtr ma przepuszczać sygnał (pasmo przepustowe – ang. *passband*), dla jakich ma tłumić (pasmo zaporowe – ang.

⁴ Chyba że $r > 1$, wtedy będzie narastająca – ale układów z takimi biegunami zazwyczaj unikamy.

stopband), a dla jakich jest nam obojętne co będzie (pasmo przejściowe – ang. *transition band*).

- Opisać dopuszczalne tolerancje: zafalowania (ang. *ripple*), minimalne tłumienie (ang. *attenuation*) – patrz rys. 4.4.
- Nie przesadzać z precyzowaniem wymagań – dać metodom projektowania trochę swobody (np. pasmo przejściowe). Zadanie projektowania filtra zbyt idealnego może skończyć się niepowodzeniem albo niepotrzebną komplikacją implementacji.
- Zdecydować czy zależy nam na liniowości fazy (stałości opóźnienia grupowego).



Rysunek 4.4. Opis tolerancji charakterystyki filtra

Aproxymacja charakterystyki

Chcemy znaleźć „najlepszą” funkcję wymierną

$$\frac{b_0 + b_1 z^{-1} + \dots + b_M z^{-M}}{1 + a_1 z^{-1} + \dots + a_N z^{-N}} \quad (\text{NOI})$$

albo

$$b_0 + b_1 z^{-1} + \dots + b_M z^{-M} \quad (\text{SOI}),$$

która dobrze wypełni nasze wymagania. Nie należy zapominać, że chcemy zaprojektować filtr stabilny i przyczynowy⁵.

Innymi słowy, projektując filtr rozwiązujemy zagadnienie aproksymacji wielomianami lub funkcjami wymiernymi. Niestety, wielomian opisujący funkcję $A(\theta)$ dla filtra SOI jest wielomianem trygonometrycznym ($H(z)$ po podstawieniu $z = e^{j\theta}$), a dla filtra NOI oba wielomiany $B(z)$ i $A(z)$ są trygonometryczne. Takie zagadnienie (w przeciwieństwie do zwykłej aproksymacji wielomianowej) nie jest łatwe do rozwiązania. Dlatego chętnie stosuje się tu metody optymalizacyjne, polegające – w dużym uproszczeniu – na heurystycznym poszukiwaniu współczynników dających najlepsze przybliżenie żądanej⁶ charakterystyki. W niniejszym ćwiczeniu

⁵ Przyczynowość jest ważna, jeśli filtr ma pracować w czasie rzeczywistym. Jeżeli filtrujemy zarejestrowany sygnał, możemy użyć filtra nieprzyczynowego – bieg czasu w sygnale nie jest związany z czasem fizycznym wykonania obliczeń.

⁶ Słowo „żądana” można też napisać bez polskich liter: „zadana” i, o dziwo, sens się nie zmienia.

spróbujemy jednak pokazać kilka metod algorytmicznych, tj. takich, w których po wykonaniu znanej liczby kroków na pewno osiągniemy sukces.

Opóźnienie grupowe

Opóźnienie grupowe jest to parametr mówiący o tym, jak opóźniana jest obwiednia impulsu o danej częstotliwości. W wielu zastosowaniach ważne jest, aby różne składowe przechodziły przez układ z tym samym opóźnieniem – inaczej następuje dyspersja sygnału, która jest zjawiskiem niepożądanym:

- dyspersja w torze telekomunikacyjnym może powodować interferencję między-symbolową i inne błędy demodulacji transmisji cyfrowej;
- dyspersja w torze audio powoduje spadek jakości odsłuchu.

Opóźnienie grupowe matematycznie definiuje się jako wziętą ze znakiem minus pochodną charakterystyki fazowej względem częstotliwości:

$$\tau(\theta) = -d\varphi(\theta)/d\theta. \quad (4.14)$$

Można udowodnić, że filtr, którego odpowiedź impulsowa jest symetryczna względem pewnego punktu P , takiego, że $2P$ jest liczbą całkowitą ($2P \in \mathbb{Z}$)

$$h(P + n) = h(P - n) \quad (4.15)$$

lub antysymetryczna

$$h(P + n) = -h(P - n), \quad (4.16)$$

będzie miał stałe opóźnienie grupowe (i wyniesie ono P – czyli może być całkowite lub połówkowe).

Prosty dowód pozostawiamy Czytelnikowi⁷.

Decyzja: SOI czy NOI?

Skrótowo podamy tu wady (–) i zalety (+) obu typów filtrów.

Filtry SOI (FIR):

- + łatwo zaprojektować metodą algorytmiczną,
- + zawsze są stabilne,
- + łatwo w nich zapanować nad zakresem dynamicznym sygnału,
- + typowe metody ich projektowania gwarantują stałe opóźnienie grupowe,
- + pozwalają stosunkowo łatwo zrealizować charakterystykę pasmowozaporową,
- wymagają wysokich rzędów do osiągnięcia wąskich pasm przejściowych.

Filtry NOI (IIR):

- + pozwalają osiągnąć wymaganą charakterystykę znacznie mniejszym kosztem (obliczeń i pamięci),
- + stosunkowo łatwo w nich implementować charakterystykę pasmowoprzepustową,

⁷ Wystarczy zacząć od nieprzyczynowego filtra z $P = 0$ i można natychmiast zobaczyć, co wynika z tej symetrii.

- niosą ze sobą ryzyko niestabilności,
- są wrażliwe na efekty kwantowania współczynników,
- wymagają dodatkowych zabiegów w celu ograniczenia ryzyka przekroczenia zakresu liczbowego,
- są trudne w projektowaniu,
- nie mogą zapewnić stałego opóźnienia grupowego.

SOI: metoda obcięcia odpowiedzi impulsowej (szeregu Fouriera)

Metoda obcięcia odpowiedzi impulsowej jest prosta, po przeprowadzeniu projektowania zgodnie z podanym „przepisem” uzyskujemy dobry wynik, wykonując z góry określoną liczbę operacji arytmetycznych.

Metodę pokażemy na przykładzie filtru dolnoprzepustowego z pasmem przepustowym do θ_p .

- Zaczynamy od idealnego filtru: $A_0(\theta) = \begin{cases} 1 & \text{dla } |\theta| < \theta_p \\ 0 & \text{dla } \theta_p < |\theta| \leq \pi \end{cases}$ i o zerowej⁸ fazie
- czyli $H_0(e^{j\theta}) = A_0(\theta)$.
- Odpowiedź impulsową obliczymy z odwrotnej transformaty Fouriera:

$$h_0(n) = \frac{1}{2\pi} \int_{-\pi}^{\pi} H_0(e^{j\theta}) e^{jn\theta} d\theta = \frac{\theta_p}{\pi} \frac{\sin n\theta_p}{n\theta_p} = \frac{1}{\pi} \cdot \frac{\sin n\theta_p}{n}$$

i otrzymujemy idealny, ale **nieprzyczynowy** filtr o **nieskończonej** odpowiedzi...

- Przycinamy więc odpowiedź, aby była skończona: $h_P[n] = h_0[n] g[n]$ (funkcja $g[n] = 0$ dla $|n| > P$ – czyli okno, np. prostokątne).
 - Aby filtr był przyczynowy, wystarczy przesunąć odpowiedź impulsową (opóźnić sygnał o P próbek), tak, aby lewa skrajna próbka znalazła się w $n = 0$: $h[n] = h_P[n - P]$.
- Ostatecznie otrzymamy

$$H(z) = \sum_{n=0}^{2P} h(n) z^{-n} = z^{-P} H_P(z).$$

Jest to filtr o liniowej charakterystyce fazowej – a więc o stałym opóźnieniu grupowym równym P (z poprzedniego rozdziału pamiętamy, że to jest ważne w wielu zastosowaniach np. telekomunikacyjnych – filtrowane impulsy nie ulegną dyspersji).

Jak ma się taki „obcięty” filtr do filtru idealnego?

Aby odpowiedzieć na to pytanie, znów uciekniemy się do mnożenia w dziedzinie czasu i równoważnego splatania w dziedzinie widma. Otóż, obcinając odpowiedź impulsową *pomnożyliśmy ją* przez okno prostokątne $g[n]$ o długości $L = 2P + 1$.

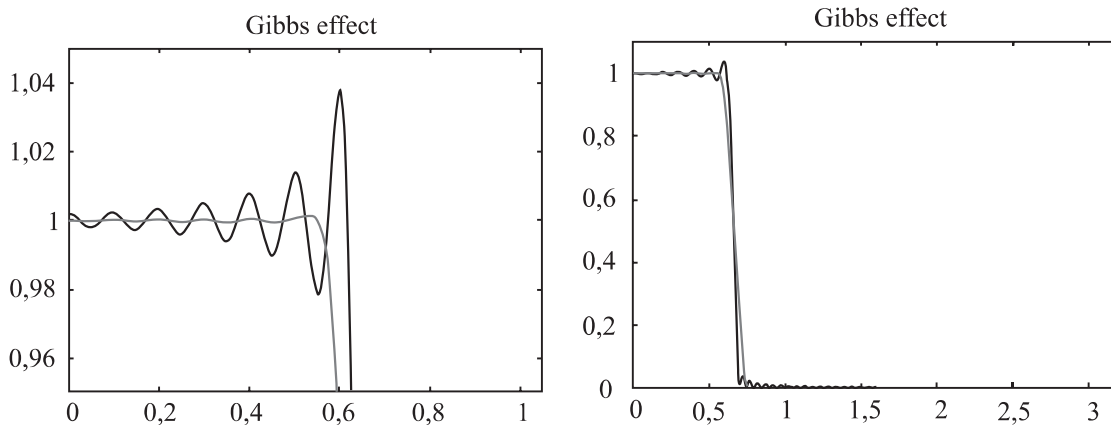
⁸ Jak coś jest stale równe zero, to jest liniową funkcją dowolnej zmiennej – np. zmiennej θ .

To znaczy, że w dziedzinie widma *spletliśmy* idealną charakterystykę z funkcją $G(\theta) = \frac{\sin L\theta/2}{\sin \theta/2}$.

Taki splot:

- rozmył nam ostre przejście z pasma przepustowego do zaporowego – uzyskało ono teraz praktycznie szerokość równą szerokości listka głównego $G(\theta)$,
- przeniósł nieco energii z pasma przepustowego do zaporowego, splatając z listkami bocznymi $G(\theta)$ – skutkiem są zafalowania w pasmie przepustowym i niepełne tłumienie w pasmie zaporowym (w obu przypadkach zobaczymy tu kształt listków bocznych – jest to znany z syntezy sygnału prostokątnego efekt Gibbsa).

Efekt zafalowań możemy zmniejszyć, używając innego okna (rys. 4.5) – ale kosztem będzie szersze pasmo przejściowe (bo „lepsze” okna mają szerszy listek główny $G(\theta)$).



Rysunek 4.5. Efekt Gibbsa (linia czarna) i jego zmniejszenie z użyciem okna Hamminga (linia szara)

Warto podkreślić, że obcięcie oknem prostokątnym prowadzi do optymalnej aproksymacji w sensie średniokwadratowym (która jednak nie jest optymalna w sensie zdrowego rozsądku).

SOI i NOI – metody optymalizacyjne

Metody optymalizacyjne projektowania filtrów zazwyczaj pozwalają spełnić wymagania przy niższym rzędzie filtru, niż metody „deterministyczne”. Za to obarczone są ryzykiem długich obliczeń w czasie projektowania i ewentualnego niezyskania dobrego rozwiązania.

Metody optymalizacyjne rozwiązywania wszelkich zagadnień polegają na sformułowaniu funkcji celu i znalezieniu jej maksimum – lub odwrotnie, sformułowaniu funkcji kary i szukaniu minimum. Nie jest celem tego przedmiotu wykład metod, jakimi się to osiąga; ograniczymy się do pokazania, jak taką funkcję kary można sformułować.

Otóż celem przy projektowaniu filtru jest uzyskanie charakterystyki $A(\theta)$ możliwie bliskiej charakterystyce idealnej $A_0(\theta)$ – oznacza to, że chcielibyśmy minimalizować odchyłkę $|A(\theta) - A_0(\theta)|$ na całym interesującym nas przedziale wartości θ (możemy np. wykluczyć z minimalizacji pasmo przejściowe). Aby zastosować me-

tody optymalizacji, trzeba ten nasz cel sprowadzić do minimalizacji pewnej wartości skalarnej ε , charakteryzującej odchyłkę na całym przedziale.

Ponieważ do minimalizacji zatrudnimy komputer, odchyłkę będziemy obliczać w skończonej liczbie punktów z tego przedziału (przy czym niektóre metody zmieniają rozkład tych punktów w trakcie iteracji). Popularne podejście zwane *średniokwadratowym* polega na sumowaniu kwadratów odchyłek:

$$\varepsilon = \sum_k |A(\theta_k) - A_0(\theta_k)|^2. \quad (4.17)$$

Podejście średniokwadratowe często prowadzi do efektywnych algorytmów minimalizacji. Warto jednak przypomnieć sobie przykład efektu Gibbsa: zachodzi on właśnie dla najlepszego średniokwadratowo przybliżenia funkcji prostokątnej; mimo że średniokwadratowy błąd jest mały, to maksymalny błąd pozostaje duży.

Dlatego w projektowaniu filtrów często stosuje się podejście zwane *minimaksowym* – minimalizuje się w nim maksymalną odchyłkę. Ponieważ to prowadzi do nieciekawych własności funkcji kary⁹, często przybliżamy tę funkcję sumą błędów podniesionych do wysokiej potęgi (duże wartości p , a często – p narastające w kolejnych iteracjach):

$$\varepsilon = \sum_k |A(\theta_k) - A_0(\theta_k)|^{2p}. \quad (4.18)$$

Praktycznie taką metodykę realizują gotowe funkcje Matlaba:

- funkcja `firpm` pozwala zaprojektować filtr SOI metodą minimaksową¹⁰,
- funkcja `yulewalk` aproksymuje średniokwadratowo wymaganą charakterystykę dla filtru NOI.

NOI – projektowanie za pomocą prototypu analogowego

Poniższa metoda projektowania pozwala uzyskać stosunkowo dobre wyniki przy deterministycznym nakładzie obliczeń.

We wstępnych rozważaniach zauważyliśmy, że potrzebujemy niełatwej aproksymacji zadanej funkcji ilorazami wielomianów trygonometrycznych. Dla filtrów analogowych opracowano wiele metod *łatwej* aproksymacji funkcji zwykłymi funkcjami wymiernymi. Tu spróbujemy ten olbrzymi zasób wiedzy wykorzystać, projektując filtr analogowy, a następnie przenosząc projekt do dziedziny filtrów cyfrowych.

Zasadniczymi różnicami między pojęciem częstotliwości dla sygnałów z czasem dyskretnym w stosunku do sygnałów z czasem ciągłym są:

- unormowanie częstotliwości do częstotliwości próbkowania,
- okresowość osi częstotliwości.

⁹ Dobrze jest np. jeśli funkcja kary jest różniczkowalna – po szersze informacje odsyłamy do literatury na temat optymalizacji.

¹⁰ Literki „pm” pochodzą od nazwisk autorów tego algorytmu – Parksa i McClellana.

To właśnie okresowość częstotliwości jest przyczyną, dla której charakterystyki częstotliwościowe filtrów cyfrowych są opisane *trygonometrycznymi* funkcjami wymiernymi.

Aby ułatwić zrozumienie toku postępowania przy projektowaniu filtru NOI za pomocą prototypu analogowego, najpierw nakreślimy plan działania, a w kolejnych akapitach wyjaśnimy dokładniej kolejne kroki. Aby zaprojektować filtr tą metodą, musimy:

- przygotować wymagania na filtr cyfrowy w dziedzinie częstości unormowanej θ ,
- przetłumaczyć te wymagania na prototyp analogowy, a więc na dziedzinę ω ,
- zaprojektować transmitancję prototypu jako funkcję wymierną $H(s)$,
- przetłumaczyć $H(s)$ na wymierną funkcję trygonometryczną $H(z)$.

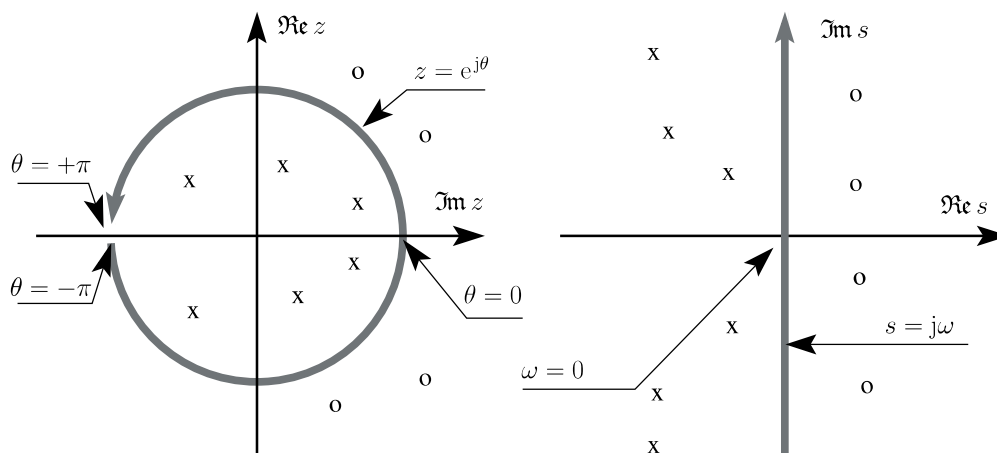
Na szczęście operacje te Matlab dostarcza w postaci gotowych funkcji. Tutaj więc tylko spróbujemy zrozumieć na czym one polegają od strony pojęciowej.

Zacznijmy od końca – od zamiany transmitancji filtru analogowego $H(s)$ na transmitancję filtru cyfrowego $H(z)$. Chcemy przy tym przekształcić jeden okres osi pulsacji θ (od $-\pi$ do π) na całą oś pulsacji ω (od $-\infty$ do $+\infty$), uwzględniając przy tym okres próbkowania T_s . Bardzo potrzebne też byłoby, gdyby ponadto zagwarantować stabilność uzyskanego filtru.

Takie właściwości ma tzw. *przekształcenie biliniowe* przeprowadzające całą dziedzinę z na dziedzinę s

$$s = \frac{2}{T_s} \frac{1 - z^{-1}}{1 + z^{-1}}. \quad (4.19)$$

Przekształcenie to przeprowadza oś pulsacji $z = e^{j\theta}$ na oś pulsacji $s = j\omega$, i wewnątrz okręgu jednostkowego $|z| < 1$ na lewą półpłaszczyznę $\Re s < 0$. Filtr zaprojektowany jako stabilny w s zostanie więc przekształcony na filtr stabilny w z (patrz rys. 4.6).



Rysunek 4.6. Przekształcenie biliniowe

Jeśli, korzystając z powyższego przekształcenia, zaprojektujemy transmitancję filtru analogowego $H(s)$ i następnie podstawimy s według wzoru (4.19), to

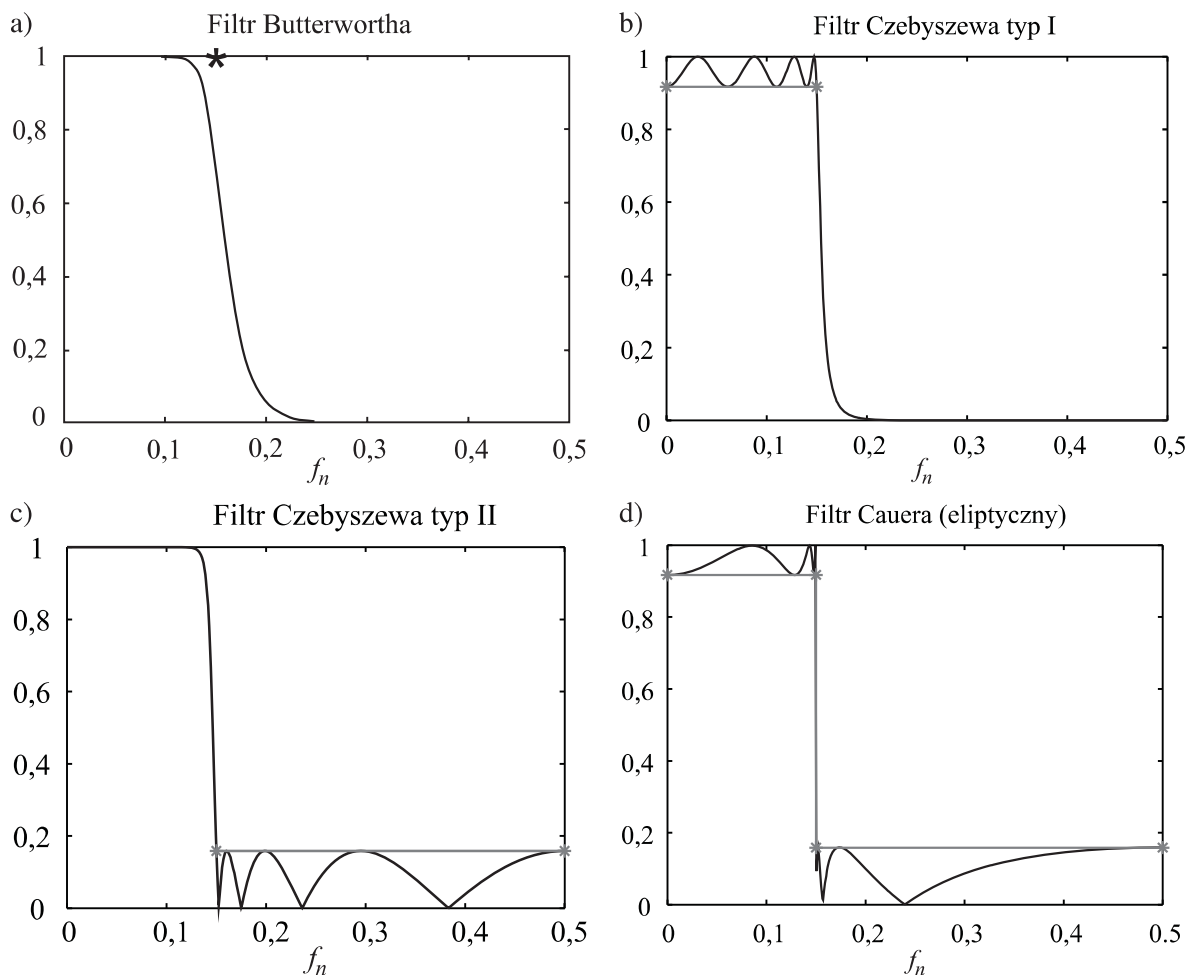
charakterystyka częstotliwościowa filtru ulegnie pewnemu zniekształceniu – pulsacji analogowej ω odpowiadać będzie pulsacja θ taka, że $j\omega = \frac{2}{T_s} \frac{1-(e^{j\theta})^{-1}}{1+(e^{j\theta})^{-1}}$. Po niewielkim uproszczeniu¹¹ widać, że

$$\omega = \frac{2}{T_s} \operatorname{tg} \frac{\theta}{2}. \quad (4.20)$$

Wobec tego wymagania na filtr (np. pulsację graniczną θ_b) trzeba przenieść do dziedziny ω właśnie według powyższego wzoru. Należy tylko pamiętać, że funkcja opisana wzorem (4.20) jest nieliniowa¹².

Procedura projektowania filtru cyfrowego przy zastosowaniu przekształcenia biliniowego polega więc na:

– przeliczeniu wymaganych unormowanych częstotliwości granicznych w dziedzinie



Rysunek 4.7. Poglądowe charakterystyki amplitudowe dolnoprzepustowych filtrów zaprojektowanych z wykorzystaniem aproksymacji: a) Butterwortha, b), c) Czebyszewa typ I i II, d) Cauera

¹¹ Wystarczy podstawić $1 = e^{+j/2\theta} \cdot e^{-j/2\theta}$ i $(e^{j\theta})^{-1} = e^{-j/2\theta} \cdot e^{-j/2\theta}$, a potem wyciągnąć $e^{-j/2\theta}$ przed nawias.

¹² Po angielsku nazywa się tę operację *prewarping*; czasownik *to warp* oznacza „wykrzywiać, wypaczać”.

- czasu dyskretnego na dziedzinę czasu ciągłego (wzór (4.20) – po angielsku nazywa się tę operację terminem *prewarp*),
- zastosowaniu gotowych metod aproksymacji funkcjami opartymi na znanych wielomianach i obliczeniu transmitancji prototypu $H_a(s)$,
 - podstawieniu zamiast s wyrażenia biliniowego (4.19).

Korzystając z gotowych funkcji Matlaba realizujących tę procedurę, warto znać właściwości typowo stosowanych do aproksymacji rodzin wielomianów:

- wielomiany Butterwortha dają aproksymację maksymalnie płaską (bez zafalowań)¹³,
- wielomiany Czebyszewa dają zafalowania w jednej części (typ I – w pasmie przepustowym, II – w zaporowym), a płaską charakterystykę w drugiej,
- wielomiany Cauera (eliptyczne) dają zafalowania w obu pasmach.

Poglądowo aproksymacje te (dla dziedziny ciągłej) przedstawiono na rys. 4.7. Limity zadane podczas projektowania zaznaczono szarymi odcinkami i gwiazdkami.

4.1.4. Triki implementacyjne

Problemy implementacyjne zostaną tu tylko zasygnalizowane skrótowo.

Oszczędne implementacje filtrów SOI

Spełnienie wymagań przez filtry SOI często wymaga użycia wysokich rzędów filtru. Jest jednak kilka sposobów na zmniejszenie złożoności obliczeniowej filtru SOI.

Po pierwsze, w filtrze o liniowej charakterystyce fazowej można wykorzystać symetrię odpowiedzi impulsowej – najpierw zsumować dwie próbki sygnału, a potem raz je pomnożyć przez odpowiedni współczynnik, i w ten sposób oszczędzić połowę mnożeń.

Po drugie, jeśli odpowiedź impulsowa filtru jest długa, warto zastąpić splot w dziedzinie czasu mnożeniem w dziedzinie widma – FFT i IFFT są algorytmami bardzo efektywnymi. Trzeba tylko pamiętać, że w ten sposób realizujemy splot kołowy – aby był on równoważny liniowemu, trzeba dopełnić sygnały zerami (a potem odpowiednio sklejać wyniki filtrowania kolejnych bloków)¹⁴. Trzeba też pamiętać, że tu *nie używa się okien*, bo to zniekształciłoby przebieg sygnału w dziedzinie czasu.

Po trzecie, w niektórych sytuacjach (filtr dolnoprzepustowy, uśredniający bardzo wiele próbek) można wykorzystać kaskadę filtrów grzebieniowych, w których nie potrzeba mnożenia (współczynniki są równe 0 lub 1)¹⁵.

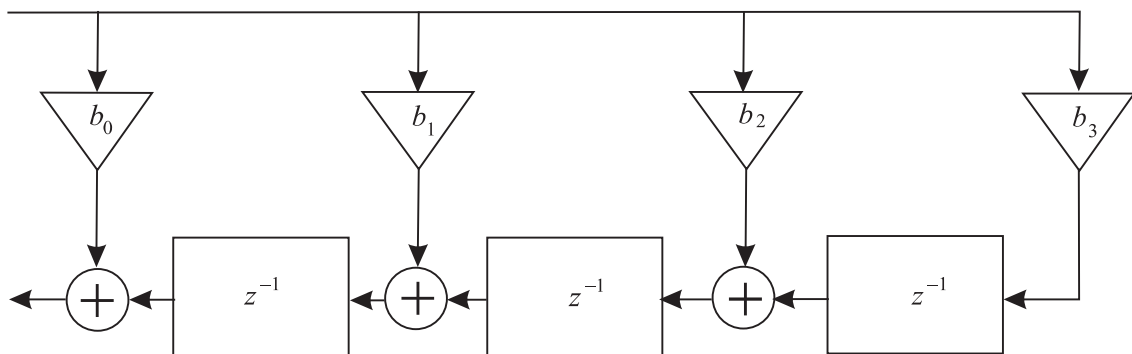
¹³ Ciekawym zbiegiem okoliczności „butter” oznacza „masło”, a masło jest gładkie.

¹⁴ Więcej informacji można szukać pod hasłem „overlap-add” i „overlap-save” – to są nazwy dwóch metod sklejanego bloków danych filtrowanych w taki sposób.

¹⁵ Ten temat jest tylko zasygnalizowany – można poszukać w literaturze lub na wykładzie z przedmiotu TRA.

W strukturze transwersalnej (czyli wprost według definicji splotu (4.1)), pokazanej na początku tego rozdziału, na rys. 4.1, wszystkie sumowania wykonuje się kaskadowo. Nie ma to raczej znaczenia w implementacji programowej na pojedynczym procesorze, natomiast w implementacjach sprzętowych (albo masywnie wieloprocessorowych) może być krytycznym czynnikiem ograniczającym prędkość przetwarzania przy wysokich rzędach filtru – dane muszą przepropagować przez wszystkie sumatory w jednym cyklu zegara. W pewnym sensie każdy sumator „czeka” na wynik z poprzedniego.

Jeśli zamienimy miejscami bloki (rys. 4.8), to każdy sumator ma na wykonanie sumowania cały cykl – minus czas mnożenia przez współczynnik. Innymi słowy, sumatory pracują równolegle, nie czekając na siebie wzajemnie.



Rysunek 4.8. Filtr SOI (FIR) w strukturze odwróconej

Problem dynamiki

Po zaprojektowaniu filtru należy rozważyć, jak duże liczby mogą się pojawić wewnątrz układu filtru (np. po pomnożeniu sygnału przez duży współczynnik, albo po zakumulowaniu próbek w pętli sprzężenia zwrotnego w filtrze NOI) – przekroczenie zakresu reprezentacji liczbowej grozi nieprzyjemnymi efektami, które bardzo trudno zanalizować.

Ryzyko takich efektów można zmniejszyć, dzieląc filtr na sekcje niższego rzędu i odpowiednio ustawiając kolejność sekcji oraz ich wzmocnienia.

Sekcja bikwadratowa

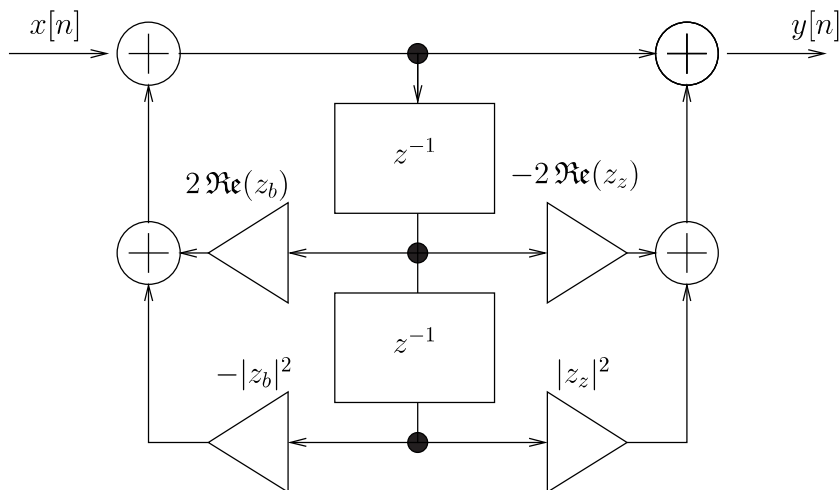
Zastanówmy się, jak można filtr wysokiego rzędu podzielić na sekcje niższych rzędów. Matematycznie polega to na zapisaniu licznika i mianownika w postaci iloczynowej, co pozwala następnie podzielić zera i bieguny na grupy realizowane w różnych sekcjach.

Najmniejszą nietrywialną sekcją jest sekcja rzędu pierwszego $H_1(z) = \frac{z-z_z}{z-z_b}$. Pamiętajmy, że zera i bieguny transmitancji mogą być zespolone – w takiej sytuacji mielibyśmy kłopot z implementacją: mimo że współczynniki filtru i sygnały są rzeczywiste, trzeba by korzystać z arytmetyki liczb zespolonych. Dlatego najczęściej wykorzystuje się fakt, że jeśli wielomian o współczynnikach rzeczywistych ma

pierwiastki zespolone, to występują one parami sprzężonymi. To oznacza, że warto takie pary zebrać razem, i wtedy otrzymuje się sekcję rzędu drugiego:

$$H_{biq}(z) = \frac{z - z_z}{z - z_b} \cdot \frac{z - z_z^*}{z - z_b^*} = \frac{1 - 2 \Re(z_z)z^{-1} + |z_z|^2 z^{-2}}{1 - 2 \Re(z_b)z^{-1} + |z_b|^2 z^{-2}} \quad (4.21)$$

zwaną sekcją bikwadratową.



Rysunek 4.9. Implementacja sekcji bikwadratowej w strukturze odwróconej – oszczędzającej zasoby pamięciowe

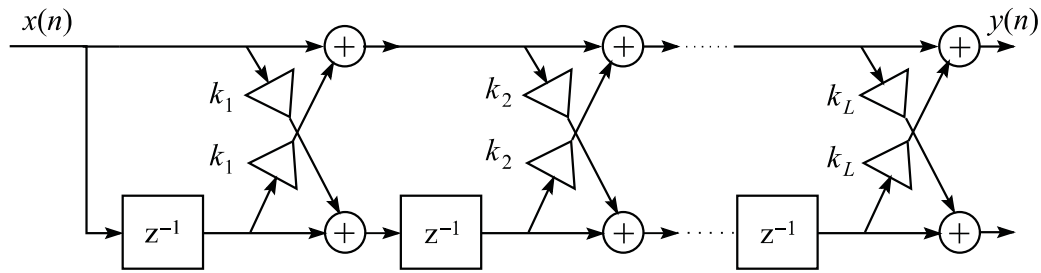
Problem stabilności

Filtr NOI może być niestabilny z kilku powodów.

- Procedura projektowania (np. optymalizacyjna) może dać w wyniku biegun poza okręgiem jednostkowym – wtedy można zareagować, przenosząc ten biegun do wewnątrz $p := 1/p$, co nie zmienia charakterystyki amplitudowej.
- Kwantowanie wartości współczynników może „wyprowadzić” biegun, który leżał blisko okręgu jednostkowego poza tę granicę. Radą na to jest dzielenie filtru na sekcje niskiego rzędu (najczęściej rzędu 2, czyli bikwadratowe) – wtedy zmniejsza się skala wartości współczynników. Inną metodą jest zastosowanie struktury kratowej filtru, w której wrażliwość na błędy współczynników jest mniejsza (za cenę większej złożoności obliczeniowej).

Struktura kratowa (ang. *lattice*)

Struktura kratowa (rys. 4.10 – wersja SOI) jest bardzo często używana w filtrach adaptacyjnych. Tematykę filtrów kratowych studenci mogą zgłębiać na przedmiotach zaawansowanych. Wspomnimy tylko, że w strukturze tego typu realizuje się różne filtry (SOI i NOI), i że obliczanie ich współczynników nie jest bardzo proste.

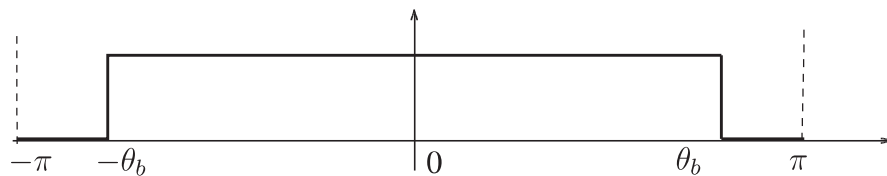


Rysunek 4.10. Struktura kratowa filtru SOI

4.2. Zadania do pracy własnej studenta

Podobne zadania mogą znaleźć się na wejściówce. Nie dotyczy to zadań oznaczonych tu jako „trudne”.

1) Idealny filtr LP.



Oblicz odpowiedź impulsową idealnego filtru dolnoprzepustowego o charakterystyce amplitudowej jak na rysunku. Przyjmij $\theta_b = \pi/4$ i charakterystykę fazową równą zero. **Wskazówka:** Przecałkowanie funkcji wykładniczej w zadanych granicach nie jest trudne.

2) Obcinając wynik poprzedniego zadania oblicz współczynniki przyczynowego filtru SOI rzędu 8 (czyli mającego odpowiedź impulsową o długości 9 próbek). Naszkicuj odpowiedź impulsową, zastanów się jakie będzie on miał opóźnienie grupowe.

Wskazówka: Zajrzyj do opisu metody projektowania filtrów „poprzez obcięcie szeregu Fouriera”.

3) Myśląc kategoriami „mnożenie w czasie = spłot w częstotliwości” spróbuj naszkicować charakterystykę uzyskanego filtru przyczynowego i oszacować szerokość pasma przejściowego.

4) Jak zmieniłaby się ww. charakterystyka, gdyby zamiast obcięcia użyć okna (nieprostokątnego)?

5) Oblicz i naszkicuj charakterystykę amplitudową filtru SOI o współczynnikach:

- $[1, -1]$,
- $[1, 1, 1]$,
- $[1, -2, 1]$.

6) Przeanalizuj filtr NOI o jednym biegunie rzeczywistym w punkcie a . Wybierz $a: |a| < 1$, bo nie chcemy filtru niestabilnego.

- Zapisz jego transmitancję w dziedzinie \mathcal{Z} .
- Znajdź charakterystykę filtru, spróbuj ją naszkicować (do tego celu wylicz wartości w charakterystycznych punktach – max, min itd. ...).