

LABORATORIUM CYFROWEGO PRZETWARZANIA SYGNAŁÓW

Ćwiczenie 6

Filtracja optymalna i adaptacyjna

1. Cel ćwiczenia

- Zapoznanie się z projektowaniem filtrów optymalnych metodą minimalizacji wskaźnika jakości i algorytmami LMS i RLS adaptacji współczynników filtrów.
- Przeprowadzenie eksperymentów symulacyjnych z filtrami adaptacyjnymi w środowisku Simulink.

2. Podstawy teoretyczne

Zagadnienie estymacji (odtworzenia) użytecznego sygnału $s(n)$ na podstawie innego dostępnego sygnału $x(n)$, który jest zaszumioną wersją $s(n)$, tzn. $x(n) = s(n) + w(n)$ ($w(n)$ – niepożądane zakłócenie) jest jednym z centralnych problemów cyfrowego przetwarzania sygnałów. Trudność polega na tym, że widmo zakłócenia pokrywa się zwykle z widmem sygnału użytecznego i filtracja zakłócenia jest związana z usunięciem części sygnału użytecznego. Dlatego filtrów optymalnych nie projektuje się w dziedzinie częstotliwości, ale z wykorzystaniem statystycznych właściwości sygnału podlegającego filtracji. W ćwiczeniu będziemy zajmować się estymacją liniową przy addytywnych zakłóceniach.

2.1. Filtracja optymalna

Struktura filtra optymalnego jest przedstawiona na rys.1. Ma on dwa sygnały wejściowe: sygnał filtrowany $x(n)$ oraz sygnał odniesienia $d(n)$, i dwa sygnały wyjściowe: wynik filtracji $y(n)$, który jest estymatą sygnału odniesienia, tj. $y(n) = \hat{d}(n)$, oraz błąd estymacji $e(n) = d(n) - y(n)$. Zakładamy, że sygnały wejściowe są *stacjonarne* (w szerszym sensie). Ograniczymy się do filtrów nierekursywnych (o skończonej odpowiedzi impulsowej SOI (FIR)):

$$H(z) = h_0 + h_1 z^{-1} + h_2 z^{-2} + \dots + h_M z^{-M}, \quad (6.1)$$

ponieważ gwarantują one stabilność i dlatego są najczęściej stosowane jako filtry adaptacyjne. Wyjście filtra można zapisać w formie:

$$y(n) = \sum_{k=0}^M h_k x(n-k) = \mathbf{h}^T \mathbf{x}(n), \quad (6.2)$$

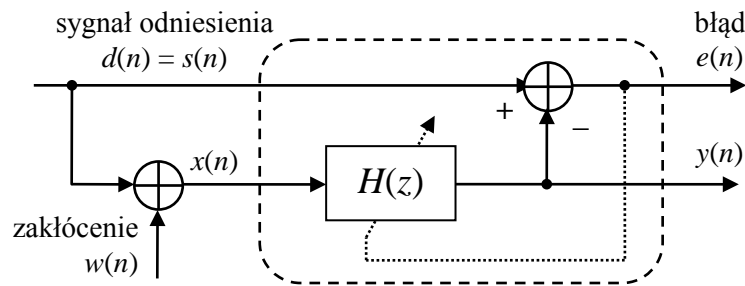
gdzie: $\mathbf{h} = [h_0, h_1, h_2, \dots, h_M]^T$, $\mathbf{x}(n) = [x(n), x(n-1), x(n-2), \dots, x(n-M)]^T$ (wskaźnik T oznacza transpozycję).

Zadaniem filtra jest takie przekształcanie sygnału wejściowego $x(n)$, aby wynik filtracji był jak najbardziej zbliżony do sygnału odniesienia przy założonym kryterium błędu. Najczęściej stosuje się kryterium minimalizacji błędu średniokwadratowego (*mean-square error* MSE)

$$J = E[e^2(n)], \quad (E[\cdot] \text{ oznacza wartość oczekiwaną}) \quad (6.3)$$

gdzie

$$e(n) = d(n) - y(n) = d(n) - \sum_{k=0}^M h_k x(n-k). \quad (6.4)$$



Rys. 1. Struktura filtra optymalnego. W przypadku stacjonarym parametry filtra $H(z)$ są wyznaczone jeden raz przez minimalizację kryterium błęd z $e(n)$

Błąd dopasowania jest funkcją współczynników filtra $\{h_k\}$:

$$J(\mathbf{h}) = E \left[\left(d(n) - \mathbf{h}^T \mathbf{x}(n) \right)^2 \right] = E[d^2(n)] - 2\mathbf{h}^T \mathbf{r}_{dx}^{(n)} + \mathbf{h}^T \mathbf{R}_{xx}^{(n)} \mathbf{h}, \quad (6.5)$$

gdzie $\mathbf{R}_{xx}^{(n)} = E[\mathbf{x}(n)\mathbf{x}^T(n)]$ jest macierzą autokorelacji sygnału wejściowego

$$\left[\mathbf{R}_{xx}^{(n)} \right]_{i,j} = R_{xx}^{(n)}(i-j) = E[x(n-i)x(n-j)], \quad i, j = 0, 1, \dots, M, \quad (6.6)$$

a $\mathbf{r}_{dx}^{(n)} = E[d(n)\mathbf{x}(n)]$ - wektorem korelacji wzajemnej pomiędzy $d(n)$ i $x(n-i)$:

$$\left[\mathbf{r}_{dx}^{(n)} \right]_i = r_{dx}^{(n)}(i) = E[d(n)x(n-i)], \quad i = 0, 1, \dots, M \quad (6.7)$$

Optymalne wartości współczynników filtra otrzymuje się w wyniku przyrównania do zera pochodnej:

$$\frac{\partial J(\mathbf{h})}{\partial \mathbf{h}} = -2\mathbf{r}_{dx}^{(n)} + 2\mathbf{R}_{xx}^{(n)} \mathbf{h} = \mathbf{0}, \quad (6.8)$$

co prowadzi do układu liniowego tzw. równań normalnych:

$$\mathbf{R}_{xx}^{(n)} \cdot \mathbf{h} = \mathbf{r}_{dx}^{(n)} \quad (6.9)$$

Rozwiązanie tych równań daje optymalne współczynniki filtra:

$$\mathbf{h}^{opt} = [\mathbf{R}_{xx}^{(n)}]^{-1} \mathbf{r}_{dx}^{(n)}, \quad \det \mathbf{R}_{xx}^{(n)} \neq 0 \quad (6.10)$$

$$\begin{bmatrix} h_0^{opt} \\ h_1^{opt} \\ \vdots \\ h_M^{opt} \end{bmatrix} = \begin{bmatrix} R_{xx}^{(n)}(0) & R_{xx}^{(n)}(1) & \cdots & R_{xx}^{(n)}(M) \\ R_{xx}^{(n)}(1) & R_{xx}^{(n)}(0) & \cdots & R_{xx}^{(n)}(M-1) \\ \vdots & \vdots & \ddots & \vdots \\ R_{xx}^{(n)}(M) & R_{xx}^{(n)}(M-1) & \cdots & R_{xx}^{(n)}(0) \end{bmatrix}^{-1} \cdot \begin{bmatrix} r_{dx}^{(n)}(0) \\ r_{dx}^{(n)}(1) \\ \vdots \\ r_{dx}^{(n)}(M) \end{bmatrix}$$

Warunek odwracalności macierzy $\det \mathbf{R}_{xx}^{(n)} \neq 0$ jest nazywany warunkiem wystarczającego pobudzenia.

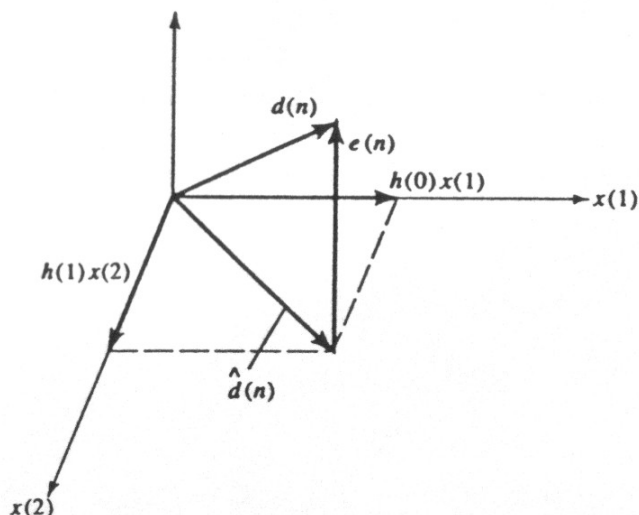
Filtr (6.10) jest nazywany filtrem Wienera. Dla filtracji optymalnej zachodzi zależność:

$$\frac{\partial J(\mathbf{h})}{\partial h_k} = 2E \left[e(n) \frac{\partial e(n)}{\partial h_k} \right] = -2E[e(n)x(n-k)] = 0, \quad k = 0, 1, \dots, M, \quad (6.11)$$

co oznacza, że filtr usuwa korelację błęd estymacji ze wszystkimi poprzednimi (i bieżącą) próbkami sygnału wejściowego:

$$r_{ex}^{(n)}(k) = E[e(n)x(n-k)] = 0. \quad (6.12)$$

Jest to tzw. właściwość ortogonalności. Jej geometryczna interpretacja jest pokazana na rys.2.



Rys. 2. Geometryczna interpretacja właściwości ortogonalności estymaty otrzymanej metodą błędu średniokwadratowego MSE dla dwóch wymiarów ($M=2$). Wyjście filtra $y(n) = \hat{d}(n)$.

2.2. Estymacja współczynników filtra optymalnego z próbek

W zagadnieniach praktycznych teoretyczne kryterium (6.3) oraz korelacje (6.6) i (6.7) wyznaczone z wartości oczekiwanych zastępuje się ich estymatami $\hat{\mathbf{R}}_{xx}$ i $\hat{\mathbf{r}}_{dx}$ obliczonymi z próbek sygnałów $x(n)$ i $d(n)$. Załóżmy, że mamy do dyspozycji po N próbek sygnału wejściowego i sygnału odniesienia:

$$\mathbf{x} = [x(0), x(1), x(2), \dots, x(N-1)]^T, \quad \mathbf{d} = [d(0), d(1), d(2), \dots, d(N-1)]^T,$$

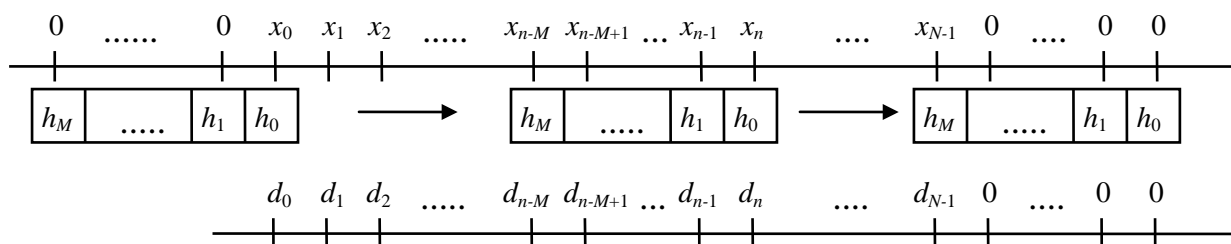
a próbki na wyjściu filtra określa równanie (6.2). Ponieważ zwykle liczba próbek jest większa niż długość filtra $N > M$, otrzymuje się *układ nadokreślony*, tzn. *liczba równań jest większa niż liczba niewiadomych* (poszukiwanych współczynników filtra). Istnieją dwie metody obliczeń w zależności od zakresu sumowania we wzorze na estymatę błędu dopasowania. Każda z nich daje inne wartości estymat $\hat{\mathbf{h}}$ współczynników filtra optymalnego.

1. Metoda autokorelacji

Kryterium błędu:

$$J_n = \sum_{n=0}^{N+M-1} e^2(n), \quad e(n) = d(n) - y(n) \tag{6.13}$$

W metodzie tej mamy $N+M$ równań na wartości wyjściowe $y(n)$. Kolejne próbki $y(n)$ są obliczane jako suma iloczynów współczynników filtra i odpowiadających im próbek $x(n)$, jak na poniższym rysunku. Jak widać, ciąg próbek sygnału wejściowego jest na krańcach uzupełniany zerami. Poza pierwszymi N równaniami w pozostałych po prawej stronie występują zera.



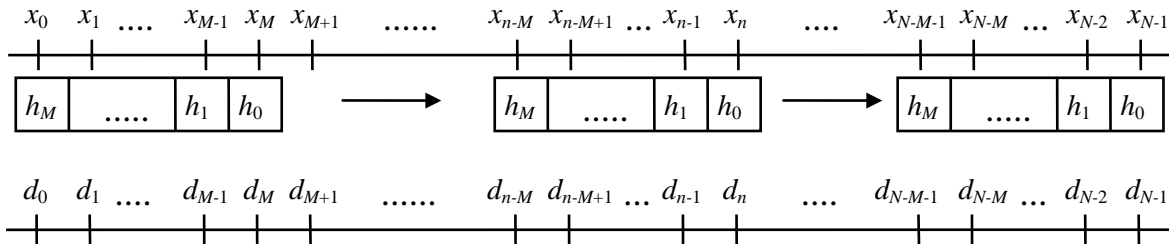
Metoda autokorelacji prowadzi do symetrycznej macierzy równań normalnych i gwarantuje, że filtr $H(z)$ jest minimalnofazowy.

2. Metoda kowariancji

Kryterium błędu:

$$J_n = \sum_{n=M}^{N-1} e^2(n), \quad e(n) = d(n) - y(n) \quad (6.14)$$

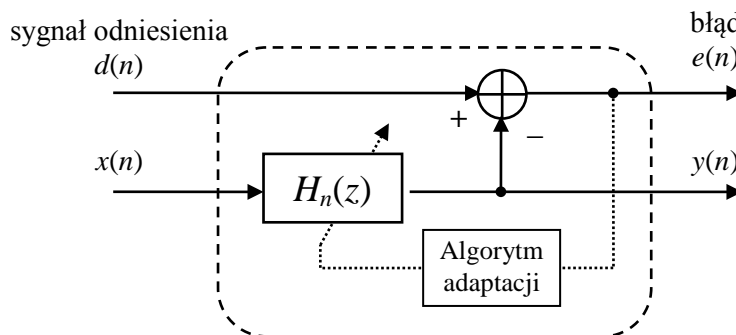
W metodzie tej mamy $N-M$ równań na wartości wyjściowe $y(n)$. Kolejne próbki $y(n)$ są obliczane jako suma iloczynów współczynników filtra i odpowiadających im próbek $x(n)$, jak na poniższym rysunku. Tworzenie równań jest ograniczone do takiego zakresu, że współczynniki filtra "nie wychodzą" poza rekord próbek wejściowych. Zwróćmy uwagę, że nie wykorzystuje się informacji z pierwszych M próbek sygnału odniesienia.



Metoda kowariancji nie gwarantuje minimalnofazowości $H(z)$.

2.3. Filtracja adaptacyjna

Filtracja adaptacyjna polega na dodaniu do struktury filtra optymalnego algorytmu adaptacji współczynników $H(z)$ (rys. 3), które są funkcjami czasu n i są na bieżąco dopasowywane w taki sposób, aby zapewniać optymalną, w sensie przyjętego kryterium błędu, filtrację sygnału wejściowego.



Rys. 3. Struktura filtra adaptacyjnego. Parametry $\mathbf{h}(n)$ filtra $H_n(z)$ są wyznaczone na bieżąco według określonego algorytmu adaptacyjnego minimalizującego kryterium błędu związane z $e(n)$

Jeżeli wszystkie sygnały w układzie są stacjonarne, to po okresie adaptacji współczynniki filtra powinny się ustalić i przyjąć wartości optymalne (warunek zbieżności procesu adaptacji):

$$\lim_{n \rightarrow \infty} \mathbf{h}(n) = \mathbf{h}^{opt} \quad (6.15)$$

W adaptacyjnym przetwarzaniu sygnałów stosuje się metody optymalizacji wieloparametrycznej (optymalizujemy $M+1$ parametrów transmitancji), które polegają poszukiwaniu minimów deterministycznych wieloargumentowych funkcji kosztu $J(\cdot)$. Najczęściej stosowane są metody gradientowe, w których modyfikacja $\Delta \mathbf{h}(n)$ jest w każdej chwili proporcjonalna do wektora ujemnego gradientu funkcji kosztu:

$$\mathbf{h}(n) = \alpha \mathbf{h}(n-1) + \Delta \mathbf{h}(n) = \mathbf{h}(n-1) - \frac{1}{2} \mu \mathbf{W}^{(n)} \cdot \nabla_n \quad (6.16)$$

$$\nabla_n = \frac{\partial J(\mathbf{h}(n))}{\partial \mathbf{h}(n)} = \left[\frac{\partial J(n)}{\partial h_0(n)}, \frac{\partial J(n)}{\partial h_1(n)}, \dots, \frac{\partial J(n)}{\partial h_M(n)} \right]^T \quad (6.17)$$

gdzie α jest tzw. współczynnikiem przeciekania (*leakage factor*, $0 < \alpha \leq 1$, typowo $\alpha=1$), $\mathbf{W}^{(n)}$ jest dodatkową macierzą wagową poprawiającą zbieżność (np. odwrotność hesjanu $\mathbf{W}^{(n)} = [\nabla_n^2]^{-1}$ w procedurze Newtona-Raphsona). Współczynnik adaptacji μ decyduje o szybkości przestrajania ($\Delta \mathbf{h}(n)$ jest proporcjonalne do μ). Ze względu na zbieżność algorytmu zachodzi warunek ograniczający:

$$0 < \mu < \frac{2}{\lambda_{\max}}, \quad (6.18)$$

gdzie λ_{\max} jest największą wartością własną macierzy autokorelacji sygnału wejściowego $\mathbf{R}_{xx}^{(n)}$.

2.4. Filtry adaptacyjne LMS

W algorytmie adaptacyjnym LMS (*Least Mean Squares*) kryterium błędu

$$J_n = e^2(n). \quad (6.19)$$

Zadaniem jest więc minimalizacja *chwilowej* (a nie oczekiwanej) wartości błędu kwadratowego, dlatego filtrację nazywa się filtracją *bez pamięci* albo *optymalizacją stochastyczną*. Jednak asymptotycznie dla $n \rightarrow \infty$ algorytm ten minimalizuje średni błąd kwadratowy.

Metoda ta obejmuje szeroką rodzinę algorytmów opisanych ogólną zależnością:

$$\mathbf{h}(n) = \alpha \mathbf{h}(n-1) + \mu(n) \mathbf{W}(n) e(n) \mathbf{x}(n) \quad (6.20)$$

Przypadki szczególne:

1. Filtr LMS

$$\mathbf{h}(n) = \alpha \mathbf{h}(n-1) + \mu e(n) \mathbf{x}(n) \quad (6.21)$$

$$e(n) = d(n) - y(n)$$

$$y(n) = \mathbf{h}^T(n-1) \mathbf{x}(n)$$

Zaletą algorytmu jest prostota i mała złożoność obliczeniowa, a wadą – wolna zbieżność algorytmu.

2. Unormowany filtr LMS (NLMS)

$$\mathbf{h}(n) = \alpha \mathbf{h}(n-1) + \mu(n) e(n) \mathbf{x}(n), \quad (6.22)$$

$$\text{gdzie } \mu(n) = \frac{\mu}{\gamma + \mathbf{x}^T(n) \mathbf{x}(n)} = \frac{\mu}{\gamma + \sum_{k=0}^M x^2(n-k)}$$

(mały parametr γ zapobiega zerowaniu się mianownika).

Filtry NLMS charakteryzują się szybszą zbieżnością i lepszą stabilnością w porównaniu z LMS.

3. Zdekorelowany filtr LMS

Zamiast $\mathbf{x}(n)$ na wejście podawany jest zdekorelowany sygnał

$$\mathbf{v}(n) = \mathbf{x}(n) - \alpha(n) \mathbf{x}(n-1), \quad \alpha(n) = \frac{\mathbf{x}^T(n) \mathbf{x}(n-1)}{\mathbf{x}^T(n-1) \mathbf{x}(n-1)} \quad (6.23)$$

który ma stosunek $\lambda_{\min}/\lambda_{\max}$ bliski jedności, co zwiększa szybkość adaptacji. $\alpha(n)$ jest współczynnikiem korelacji wektorów $\mathbf{x}(n)$ i $\mathbf{x}(n-1)$.

2.5. Filtry adaptacyjne RLS

Popularnym algorytmem adaptacji jest algorytm LS (*Least Squares*) z kryterium błędu w postaci sumy kwadratów:

$$J_n = \sum_{k=0}^n e^2(k) \quad (6.24)$$

lub ważonej sumy kwadratów WLS (*Weighted LS*):

$$J_n = \sum_{k=0}^n \lambda^{n-k} e^2(k), \quad (6.25)$$

gdzie λ jest współczynnikiem zapominania starych błędów o wartości z zakresu $0.95 \leq \lambda \leq 1$. Algorytm z wagami umożliwia adaptację do zmieniających się charakterystyk sygnałów, kiedy dane z przeszłości przestają być aktualne. Ponieważ kryterium zależy od sumy błędów, filtry LS nazywa się filtrami z

pamięcią. W trakcie adaptacji ruch trajektorii układu w przestrzeni parametrów w kierunku minimum jest mniej chaotyczny niż w przypadku metody LMS.

Równanie zmian współczynników można zapisać w postaci:

$$\mathbf{h}(n) = \alpha \mathbf{h}(n-1) + \mathbf{K}(n)e(n), \quad (6.26)$$

gdzie $\mathbf{K}(n) = [\hat{\mathbf{R}}_{xx}^{(n)}]^{-1} \mathbf{x}(n)$ nazywa się wzmocnieniem Kalmana.

Ze względu na konieczność odwracania w każdym kroku macierzy autokorelacji algorytm LS implementuje się w formie *rekurencyjnej* RLS (*Recursive LS*)

$$\mathbf{h}(n) = \mathbf{h}(n-1) + \mathbf{K}(n)e(n)$$

$$e(n) = d(n) - y(n), \quad y(n) = \mathbf{h}^T(n-1)\mathbf{x}(n)$$

$$\mathbf{K}(n) = \mathbf{P}(n)\mathbf{x}(n) = \frac{\lambda^{-1}\mathbf{P}(n-1)\mathbf{x}(n)}{1 + \lambda^{-1}\mathbf{x}^T(n)\mathbf{P}(n-1)\mathbf{x}(n)}, \quad (\text{gdzie } \mathbf{P}(n) = [\hat{\mathbf{R}}_{xx}^{(n)}]^{-1}) \quad (6.27)$$

$$\mathbf{P}(n) = \lambda^{-1}[\mathbf{I} - \lambda^{-1}\mathbf{K}(n)\mathbf{x}^T(n)]\mathbf{P}(n-1), \quad (6.28)$$

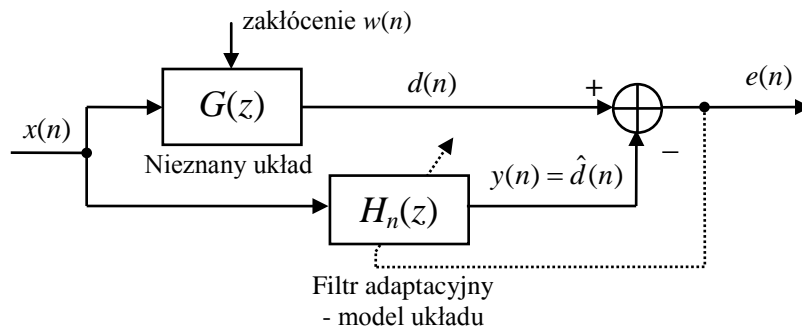
z warunkami początkowymi: $\mathbf{P}(0) = \mathbf{I} \frac{1}{\hat{\sigma}_x^2}$, $\mathbf{h}(0) = \mathbf{h}_0$.

Algorytm RLS jest bardziej złożony obliczeniowo niż LMS, ale zwykle daje szybszą i płynniejszą zbieżność.

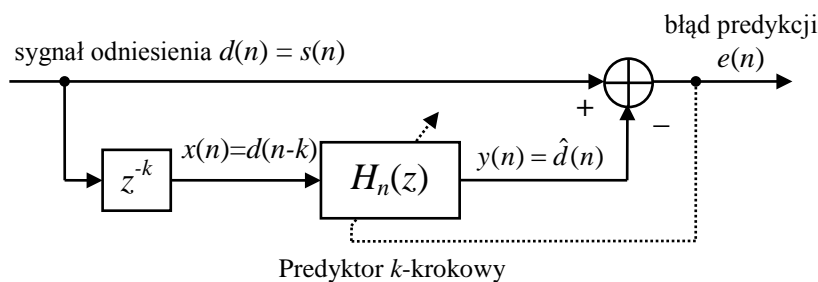
2.6. Typowe zastosowania filtracji adaptacyjnej

Poniżej przedstawiono kilka typowych struktur zastosowania filtracji adaptacyjnej ilustrujących interpretację poszczególnych sygnałów.

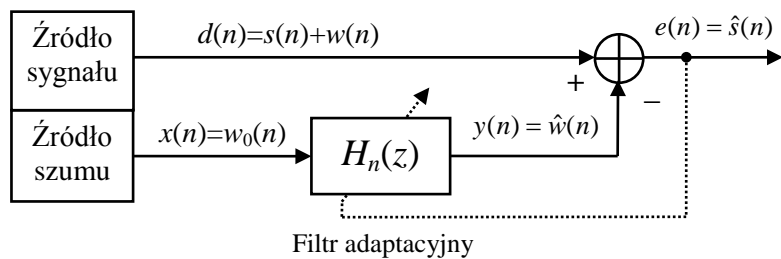
We wszystkich przypadkach wyjście filtra $y(n)$ jest estymatą tylko tej składowej sygnału odniesienia $d(n)$, która jest skorelowana z sygnałem wejściowym $x(n)$. Jeżeli $d(n) = s(n) + x_1(n)$, gdzie tylko $x_1(n)$ jest skorelowane z $x(n)$, to wyjście $y(n) = \hat{x}_1(n)$.



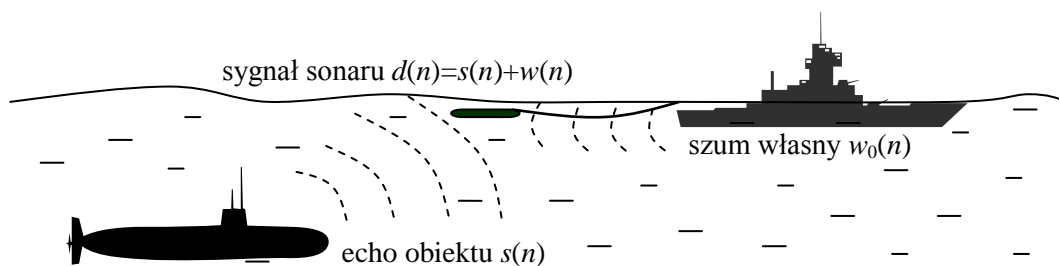
Rys. 4. Identyfikacja adaptacyjna. Transmitancja filtra adaptacyjnego jest estymatą transmitancji identyfikowanego układu $H(z) = \hat{G}(z)$.



Rys. 5. Adaptacyjna predykcja sygnału



Rys. 6. Adaptacyjne kasowanie szumu (usuwanie korelacji)



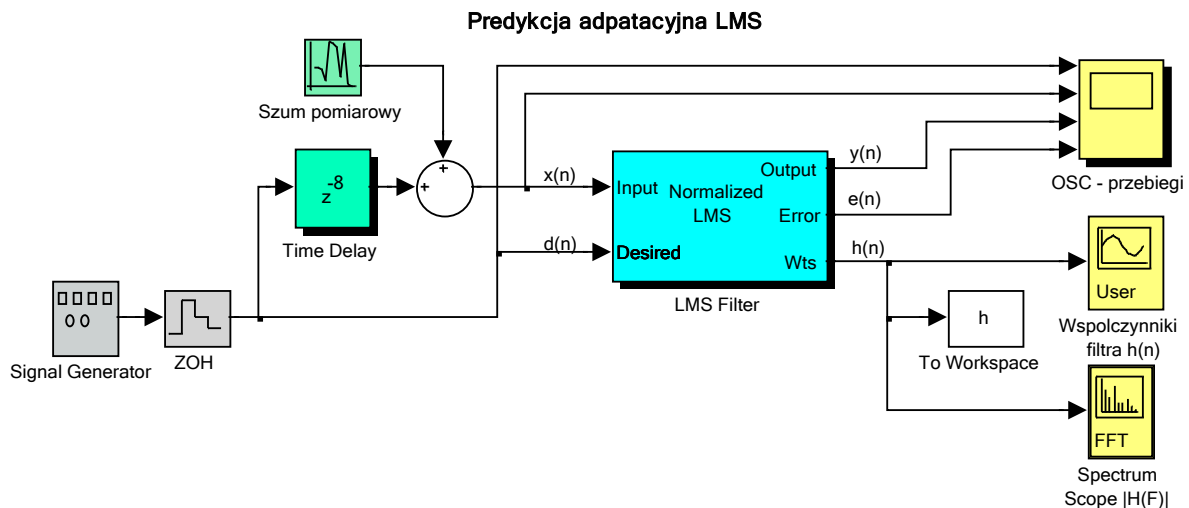
Rys. 6. Przykład zastosowania adaptacyjnego kasowania szumu. Zakłócenie skorelowane z szumem własnym okrętu dociera do sonaru jako jego słumiona i opóźniona wersja $w(n) = \alpha \cdot w_0(n - n_0)$. Zadaniem filtracji adaptacyjnej jest wytłumienie tej składowej w sygnale sonaru.

3. Symulacje komputerowe - zadania do wykonania

- Do obliczeń wykorzystywana jest biblioteka symulacyjna Simulinka *DSP System Toolbox* oraz biblioteka Matlaba *Signal Processing Toolbox*.
- Bloki instrukcji Matlaba do realizacji poszczególnych zadań wygodnie jest kopiować do edytora Matlaba i tam dokonywać edycji parametrów, dopisywania nowych linii itp., a następnie uruchamiać jako skrypty komendą **Start/Run** ▷.
- W oknie **Current Folder** przejść do folderu **Moje dokumenty/MATLAB/DSP**. Jest to folder roboczy dla ćwiczeń z *Cyfrowego przetwarzania sygnałów*.
- W folderze roboczym sprawdzić obecność modeli symulacyjnych potrzebnych do realizacji ćwiczenia: `adappredict.mdl`, `adapdelay.mdl`, `adapident.mdl`, `adapcancel.mdl`.

3.1. Liniowa predykcja sygnału

A. Otworzyć model symulacyjny `adappredict.mdl` do realizacji adaptacyjnej predykcji liniowej: `adappredict`



- Ustawić (sprawdzić) parametry symulacji w menu **Simulation | Configuration Parameters**
 Stop time=1000, Solver options: Fixed-step, discrete



Ustawić (sprawdzić) parametry bloków modelu symulacyjnego (*okno parametrów bloku otwiera się po dwukrotnym kliknięciu bloku na schemacie*):

Signal generator: Waveform: sine, Amplitude=1, Frequency=0.01 Hz,

ZOH: Sample time=1 (okres próbkowania)

Szum pomiarowy: Variance=0.0 (brak szumu), Sample time=1

Time Delay = 8 (opóźnienie sygnału wejściowego filtra w próbkach, predykcja 8-krokowa)

LMS Filter: Algorithm=Normalized LMS, Filter length=25 (FIR), Step size $\mu=0.7$, Leakage factor=1.0 (algorytm nLMS z normalizacją, długość filtra L , współczynnik adaptacji μ , współczynnik przeciekania α)

- Otworzyć okno wykresów oscyloskopu OSC i uruchomić symulację (przycisk Play ▶ lub **Simulation | Start**). Obserwować przebieg adaptacji.
Uwaga: Skale czasu i częstotliwości w oknach wykresów charakterystyki impulsowej (współczynników filtra) i amplitudowej nie są poprawnie wyskalowane i wynikają z długości filtra oraz zadanego okresu próbkowania.

- Zarejestrować przebiegi sygnałów z bloku oscyloskopu wraz z końcowymi charakterystykami impulsową $h(n)$ i amplitudową $|H(\Omega)|$ filtra predykcyjnego LMS $h(n)$ (blok Współczynniki filtra) uruchamiając w oknie komend funkcję wykreślania sygnałów zarejestrowanych w pamięci w strukturze ScopeData i wektorze h. Funkcja wykreśla: sygnał odniesienia $d(n)$, wejście filtra $x(n)$ (opóźniona i zaszumiona wersja sygnału odniesienia), wyjście filtra $y(n)=d_{est}(n)$, błąd predykcji $d(n)$, $e(n)=d(n)-y(n)$ oraz końcowe charakterystyki impulsową i amplitudową filtra adaptacyjnego.

```
plotpredict(ScopeData,h)
```

- **Uwaga:** Sprawdzić, czy plik funkcji plotpredict.m znajduje się w folderze roboczym. Jeżeli nie, podane poniżej listing funkcji należy skopiować do edytora Matlaba i zapisać w pliku o podanej nazwie w folderze roboczym. Funkcję plotpredict należy wywoływać po każdej przeprowadzonej symulacji.

```
function plotpredict(ScopeData,h)
% Rysowanie wykresów do modelu adappredict.mdl
t=ScopeData.time;
d=ScopeData.signals(1,1).values;
x=ScopeData.signals(1,2).values;
y=ScopeData.signals(1,3).values;
e=ScopeData.signals(1,4).values;
b=h(:, :, end); b=b(:);
figure(1)
subplot(411), plot(t,d), grid, ylim([-2,2]), ylabel('d(n)')
title('Predykcja adaptacyjna sygnału'),
subplot(412), plot(t,x,'m'), grid, ylim([-2,2]), ylabel('x(n)')
subplot(413), plot(t,y), grid, ylim([-2,2]), ylabel('y(n)')
subplot(414), plot(t,e,'r'), ylabel('e(n)'), xlabel('t (s)')
grid, ylim([-2,2])
figure(2)
subplot(211)
stem(0:length(b)-1,b), xlabel('n'), ylabel('h(n)')
title('Końcowe ch-ki filtra predykcji')
subplot(212)
[H,W]=freqz(b);
plot(W/pi,20*log10(abs(H)), 'r'),
xlabel('\Omega / \pi'), ylabel('|H(\Omega)| (dB)'), grid
```

- Której z podstawowych charakterystyk filtrów (LP, HP, BS, BP) odpowiada charakterystyka amplitudowa filtra predykcji i dlaczego? Zwrócić uwagę na przesunięcie fazowe sygnałów $d(n)$ i jego k -krokowej predykcji $y(n)$ względem $x(n)$.

B. Przeprowadzić obliczenia i zarejestrować wyniki dla 4 kombinacji 2 parametrów modelu:

- współczynnik adaptacji Step size $\mu = 0.7, 0.2$
- moc szumu Variance = $1e-3, 1e-1$

- Skomentować otrzymane wyniki. Jak współczynnik adaptacji i moc szumu wpływają na wielkość błędu predykcji $e(n)$ (na początku i w stanie „ustalonym”) i szybkość adaptacji (czas trwania początkowego stanu nieustalonego)?

C. Przeprowadzić eksperyment dla predykcji 15 kroków naprzód: Time Delay=15, Step size $\mu=0.2$, Variance= $1e-1$, i skomentować wyniki.

D. Zmienić algorytm filtra LMS na algorytm *bez normalizacji* (Algorithm LMS) i sprawdzić jak działa adaptacja (parametry jak w pkt. C). Zwiększyć współczynnik adaptacji do $\mu=0.5$ i 0.8 .

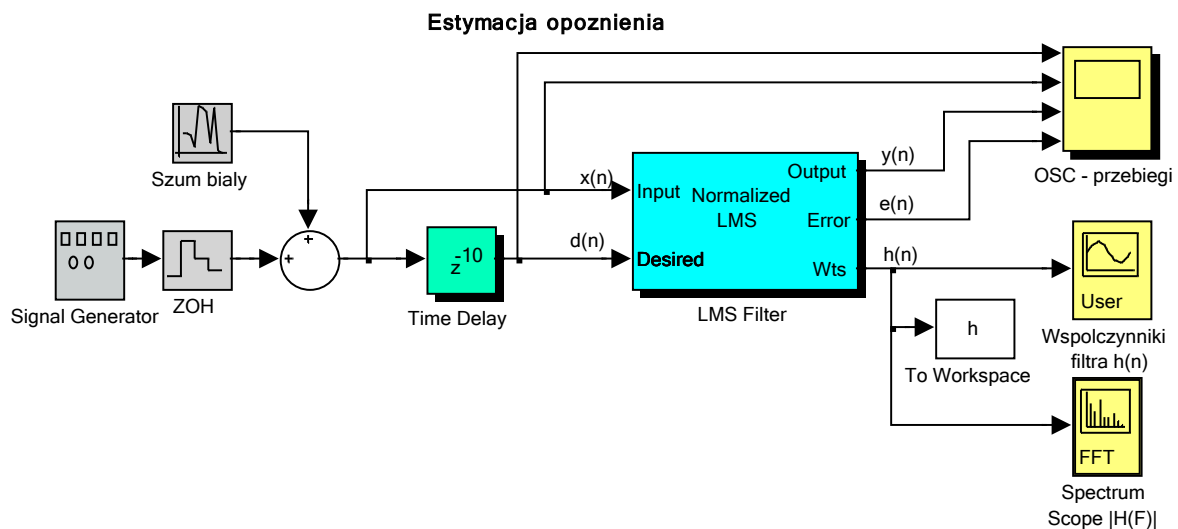
- Skomentować wyniki. Czy wystąpił brak zbieżności lub utrata stabilności algorytmu adaptacji?

3.2. Estymacja opóźnienia sygnału

A. Otworzyć model symulacyjny adapdelay.mdl do adaptacyjnej estymacji opóźnienia:

adapdelay

- Ustawić (sprawdzić) parametry symulacji w menu **Simulation | Configuration Parameters**
 Stop time=1000, Solver options: Fixed-step, discrete
 Ustawić (sprawdzić) parametry bloków modelu symulacyjnego:
 Signal generator: Waveform: sine, Amplitude=1, Frequency=0.01 Hz,
 ZOH: Sample time=1 (okres próbkowania)
 Szum: Variance=0.0 (brak szumu), Sample time=1
 Time Delay = 10 (opóźnienie sygnału odniesienia filtra w próbkach)
 LMS Filter: Algorithm=Normalized LMS, Filter length=25, Step size $\mu=0.5$, Leakage factor=1.0 (algorytm nLMS z normalizacją, długość filtra L , współczynnik adaptacji μ , współczynnik przeciekania α)



- Otworzyć okno oscyloskopu OSC, uruchomić symulację i obserwować przebieg adaptacji.
- Zarejestrować przebiegi sygnałów z bloku oscyloskopu wraz z końcowymi charakterystykami impulsową $h(n)$ i amplitudową $|H(\Omega)|$ filtra estymacji opóźnienia. Funkcja wykreśla: sygnał odniesienia $d(n)$ - sinusoida (z ewentualnym szumem białym) - opóźnione o k kroków *echo* sygnału wejściowego $x(n)$, wejście filtra $x(n)$ (sygnał bez opóźnienia), wyjście filtra $y(n)=d_{est}(n)$, błąd $e(n)=d(n)-y(n)$. Struktura funkcji jest analogiczna do struktury funkcji `plotpredict`.

`plotdelay(ScopeData, h)`

- Uwaga: Funkcję `plotdelay` należy wywoływać po każdej przeprowadzonej symulacji.
- Czy wyjście filtra dostosowało się do sygnału odniesienia bez szumu? Zwrócić uwagę na płaską *uśredniającą* odpowiedź impulsową. Któremu z podstawowych typów filtrów odpowiada końcowa charakterystyka amplitudowa filtra? W jaki sposób taki filtr estymuje opóźnienie sygnału odniesienia?
- B.** Powtórzyć obliczenia i zarejestrować wyniki dla sinusoidy z dodatkiem szumu białego o wariancji $\text{Variance}=1e-3$ i $1e-1$.
- W jaki sposób zmienia się mechanizm adaptacji w miarę wzrostu natężenia szumu? Jakie są końcowe charakterystyki filtra w przypadku szumu o większej wariancji. Co stanowi w tym przypadku estymatę opóźnienia *echa*?
- C.** Przeprowadzić analogiczne obliczenia dla sygnału z szumem i opóźnienia `Time delay=15`.
- D.** Przeprowadzić obliczenia dla `Time delay=15` z szumem o wariancji $\text{Variance}=1e-1$ zmieniając współczynnik adaptacji filtra, np. $\mu=0.2, 0.5$ i 0.9 .

➤ W jaki sposób zmiana współczynnika adaptacji wpływa na przebieg estymacji opóźnienia?

3.3. Identyfikacja układu dynamicznego

A. Otworzyć model symulacyjny adaptident.mdl do adaptacyjnej identyfikacji układu dynamicznego:

adaptident

- Ustawić (sprawdzić) parametry symulacji w menu **Simulation | Configuration Parameters**

Stop time=1500, Solver options: Fixed-step, discrete

Ustawić (sprawdzić) parametry bloków modelu symulacyjnego:

Signal generator: Waveform: square (wymuszenie prostokątne), Amplitude=1, Frequency=0.01 Hz,

ZOH: Sample time=1 (okres próbkowania)

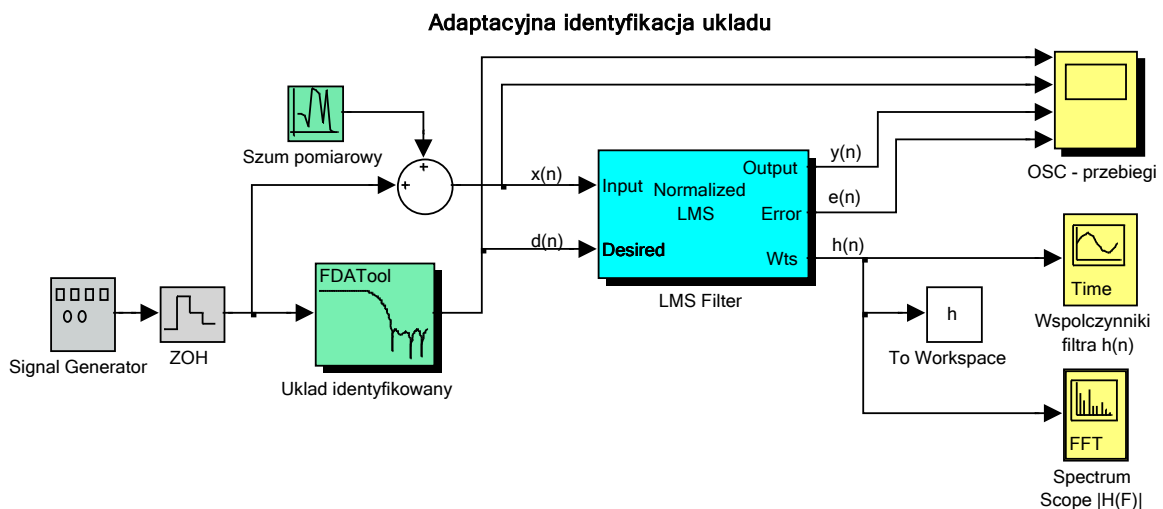
Szum pomiarowy: Variance=0.0 (brak szumu), Sample time=1



Układ identyfikowany (blok FDA Tool): Lowpass, FIR, Filter order=20, Window=Hamming, Frequency Units-Normalized (0 to 1) (częstotliwość Nyquista = 1), $\omega_c=0.5$.

Uwaga: Za każdym razem po dokonaniu ewentualnych zmian nacisnąć przycisk Design Filter.

Polecenie to wyznacza parametry transmitancji projektowanego filtra niezbędne do przeprowadzenia symulacji.



Filtr adaptacyjny: Normalized LMS, Filter length=25, Step size $\mu=0.9$, Leakage factor=1.0 (algorytm nLMS z normalizacją, długość filtra L , współczynnik adaptacji μ , współczynnik przeciekania α)



- Otworzyć okno wykresów oscyloskopu OSC, uruchomić symulację (**Play** ) i obserwować przebieg adaptacji. Automatycznego skalowania wykresów dokonuje się za pomocą przycisku **Autoscale** .
- Zarejestrować przebiegi sygnałów z bloku oscyloskopu wraz z końcowymi charakterystykami impulsową $h(n)$ i amplitudową $|H(\Omega)|$ filtra identyfikacji dynamiki układu. Funkcja wykreśla: sygnał odniesienia $d(n)$ – wyjście identyfikowanego układu, wejście filtra $x(n)$ – wejście identyfikowanego układu (z ew. szumem pomiarowym), wyjście filtra $y(n) = d_{est}(n)$ – estymata wyjścia układu identyfikowanego, błąd identyfikacji $e(n) = d(n) - y(n)$. Struktura funkcji jest analogiczna do struktury funkcji plotpredict.

plotdident (ScopeData, h)

- Uwaga: Funkcję plotident należy wywoływać po każdej przeprowadzonej symulacji.

➤ Sprawdzić zgodność końcowych charakterystyk $h(n)$ i $|H(\Omega)|$ filtra adaptacyjnego identyfikacji z rzeczywistymi charakterystykami identyfikowanego układu (filtra) w oknie bloku FDA Tool (*Filter Design and Analysis*): impulsową (przycisk  na belce menu) i amplitudową (przycisk  na belce). Zadany rząd (długość minus 1) filtra adaptacyjnego jest inny niż rząd identyfikowanego układu, który z założenia nie jest znany. Czy podobne wartości mają współczynniki odpowiedzi impulsowych o takich samych indeksach?

B. Powtórzyć obliczenia i zarejestrować wyniki z szumem zakłócenia: Szum: Variance=1e-2.


C. Zmienić identyfikowany układ na pasmowoprzepustowy:

Układ identyfikowany (blok FDA Tool): Bandpass, FIR, Filter order=20, Window=Hamming, Frequency Units-Normalized (0 to 1), wc1=0.4, wc2=0.6.

- Przeprowadzić symulację z szumem i zarejestrować wyniki.
- Czy charakterystyki (impulsowa i amplitudowa) filtra adaptacyjnego we właściwy sposób odwzorowują charakterystyki identyfikowanego układu pasmowoprzepustowego?
- Powtórzyć obliczenia dla małej wartości współczynnika adaptacji, np. $\mu=0.3$.
- Jak mniejsza wartość μ wpłynęła na przebieg adaptacji, w szczególności na wykres błędu $e(n)$?

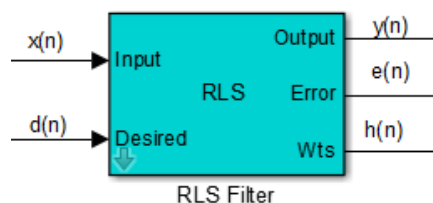
D. Ustawić długi czas symulacji Stop time=50000 i współczynnik $\mu=0.9$. Uruchomić symulację i **obserwować przebieg adaptacji przy zmienianiu typu i parametrów układu identyfikowanego on-line**. Można zmieniać np. typ układu (w bloku FDA Tool) BP → BS (Bandstop, wc1=0.4, wc2=0.6) → LP (Lowpass, wc=0.5) → HP (Highpass, wc=0.5), wariację szumu, współczynnik adaptacji μ (w bloku filtra LMS), sygnał wymuszenia (wypróbować wymuszenie sinusoidalne Waveform=sine (blok Signal generator), zamiast prostokątnego square, przy braku szumu, tj. Variance=0.0).

➤ Przeanalizować i skomentować uzyskane wyniki.

E. Zamienić filtr adaptacyjny w schemacie blokowym na filtr RLS: zaznaczyć blok filtra LMS i usunąć go, otworzyć bibliotekę bloków Simulinka **Library Browser** (przycisk  na belce okna modelu) i wybrać **DSP System Toolbox | Filtering | Adaptive filters | RLS filter** i przeciągnąć go w miejsce usuniętego bloku (Uwaga: strzałki wejść i wyjść muszą się połączyć z portami bloku!).

Parametry filtra RLS:

RLS Filter: Filter length=25 (FIR), Forgetting factor=1.0 (współczynnik zapominania λ), Initial value of weights=0.0, Initial input variance estimate=0.1



- Przywrócić wartości pozostałych parametrów jak w punkcie A (Stop time=1500, wymuszenie square, wariacja szumu =0, układ identyfikowany Lowpass). Powtórzyć obliczenia jak w punktach A, B, C i zarejestrować wyniki.
- Zaobserwować wpływ na przebieg adaptacji zmiany współczynnika zapominania na Forgetting factor=0.95 (zamiast zmiany μ dla filtra LMS jak w pkt.C).

➤ Porównać przebiegi adaptacji i efekty identyfikacji z odpowiednikami uzyskanymi dla filtra nLMS.

F. (*do uznania przez prowadzącego ćwiczenia*) Powtórzyć symulację z filtrem RLS z długim czasem symulacji jak w punkcie D i Forgetting factor=1.0. Zaobserwować efekty po wprowadzeniu zapominania zmieniając Forgetting factor filtra RLS np. na 0.99.

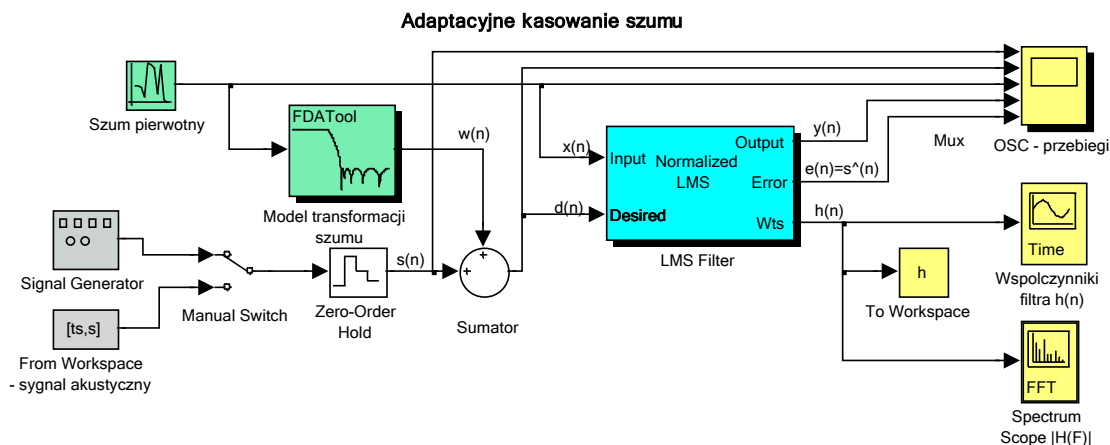
- Przeanalizować i skomentować uzyskane wyniki porównując je z wynikami otrzymanymi dla filtracji LMS.

3.4. Adaptacyjne kasowanie szumu

A. Otworzyć model symulacyjny `adapcancel.mdl` do adaptacyjnego kasowania (tłumienia) szumu dodanego, po przetworzeniu przez filtr modelujący transformację od pierwotnej postaci szumu białego, do sygnału użytecznego $s(n)$. Sygnał użyteczny z przetworzonym szumem stanowią sygnał odniesienia $d(n)$. Forma pierwotna szumu jest podawana na wejście filtrowane $x(n)$ filtra adaptacyjnego. Sygnał wyjściowy filtra $y(n)$ jest estymatą składowej sygnału odniesienia skorelowanej z sygnałem $x(n)$, czyli dodanego do sygnału odniesienia szumu. W takiej sytuacji błąd estymacji stanowi estymatę sygnału użytecznego: $e(n)=\hat{s}(n)$.

`adapcancel`



- Ustawić (sprawdzić) parametry symulacji w menu **Simulation | Configuration Parameters**
`Stop time=1500`, Solver options: Fixed-step, discrete
 Ustawić (sprawdzić) parametry bloków modelu symulacyjnego:
Signal generator: Waveform: sine (wymuszenie harmoniczne), Amplitude=1, Frequency=0.02 Hz,
ZOH: Sample time=1 (okres próbkowania)
Manual Switch: przełącznik w pozycji Signal Gen. (dwukrotne kliknięcie w obszarze bloku przełącza położenie styku).
Szum pierwotny: Variance=1e-3, Sample time=1
Model transformacji szumu(blok `FDA Tool`): Lowpass, NOI Chebyshev Type II, Filter order=10, Frequency Units-Normalized (0 to 1), wstop=0.5.
Uwaga: Za każdym razem po dokonaniu ewentualnych zmian nacisnąć przycisk **Design Filter**.
Filtr adaptacyjny: Normalized LMS, Filter length=41, Step size $\mu=0.25$, Leakage factor=1.0 (algorytm nLMS z normalizacją, długość filtra L , współczynnik adaptacji μ , współczynnik przeciekania α)



- Zainicjalizować zmienne występujące w bloku From Workspace:
`ts=0; s=0;`
- Otworzyć okno wykresów oscyloskopu OSC, uruchomić symulację i obserwować przebieg adaptacji.
- Zarejestrować przebiegi sygnałów z bloku oscyloskopu wraz z końcowymi charakterystykami impulsową $h(n)$ i amplitudową $|H(\Omega)|$ filtra kasującego szum. Funkcja wykreśla: sygnał użyteczny $s(n)$, sygnał odniesienia $d(n)$ – sygnał użyteczny z przefiltrowanym szumem pierwotnym, wyjście filtra $y(n)$ – estymata szumu w sygnale odniesienia, błąd estymacji $e(n)$ – estymata sygnału użytecznego $s(n)$. Struktura jest analogiczna do struktury funkcji `plotpredict`, zamiast szumu na wejściu filtra wykreślany jest przebieg sygnału użytecznego.

```
plotcancel(ScopeData,h)
```

- Uwaga: Funkcję `plotcancel` należy wywoływać po każdej przeprowadzonej symulacji.
- Powtórzyć symulację ze zwiększoną wariancją szumu pierwotnego: $\text{Variance}=0.3$ i 1.0 i zarejestrować wyniki.

- Jak natężenie szumu wpływa na zbieżność przebiegu adaptacji charakterystyk filtra LMS?
- Co w procesie adaptacji modeluje filtr LMS? Jak charakterystyka amplitudowa filtra adaptacyjnego ma się do widma szumu przetworzonego $w(n)$? Porównać końcowe charakterystyki filtra adaptacyjnego z charakterystykami filtra szumu dostępnymi w oknie bloku FDA Tool: impulsową (przycisk ) i amplitudową (przycisk ). Zwrócić uwagę, jak skończona odpowiedź impulsowa filtra adaptacyjnego modeluje nieskończoną odpowiedź impulsową filtra szumu. Pamiętać, że widmo mocy $P_{ww}(\Omega)$ szumu $w(n)$ jest proporcjonalne do $|H(\Omega)|^2$ filtra przetwarzającego pierwotny szum biały i że wyjście filtra ma kompensować składową szumu w sygnale odniesienia.

- B.** Przeprowadzić symulacje i analizę wyników po zmianie modelu transformacji szumu pierwotnego na filtr pasmowoprzepustowy BP dla dwóch różnych wartości współczynnika adaptacji filtra LMS i wariancji szumu (4 kombinacje).

Model transformacji szumu (blok FDA Tool): Bandpass, NOI Chebyshev Type II, Filter order=10, Frequency Units–Normalized (0 to 1), wstop1=0.2, wstop2=0.8. Uwaga: Po dokonaniu zmian nacisnąć przycisk Design Filter.

Szum pierwotny: $\text{Variance}=0.3, 1.0$

Filtr adaptacyjny: Step size $\mu=0.25, 0.8$

- C.** Dla kombinacji parametrów z pkt. B dającej najlepsze rezultaty przeprowadzić symulację i skomentować efekty dla filtracji LMS *bez normalizacji* (Algorithm: LMS w bloku filtra LMS).

- D.** Przeprowadzić eksperyment z **kasowaniem szumu w (niestacjonarnym) sygnale dźwiękowym**.

- Ustawić: 1) algorytm Normalized LMS w bloku filtra adaptacyjnego, pozostałe parametry jak w pkt. C, 2) dolnoprzepustowy model transformacji szumu w bloku FDA Tool jak w pkt. A oraz 3) wariancję szumu pierwotnego $\text{Variance}=1$.
- Wczytać (do zmiennej `s`) i odtworzyć sygnał dźwiękowy z pliku `wav` o podanej nazwie (** należy włączyć głośniki komputera **), np.

```
[s,fs,nbits]=wavread('fire'); % sygnał, częstotl. próbkow., bity/próbkę
n=length(s); % długość sygnału
ts=[0:n-1]'; % czas dyskretny
fprintf('Dane sygnału wav:\n')
fprintf('liczba próbek n = %d, cz. próbkowania fs = %d Hz, nbits = %d
bitów/próbkę\n',n,fs,nbits),
disp(' '),
sound(s,fs) % odtworzenie sygnału
```

- W oknie komend podawana jest liczba próbek sygnału, dla `fire11.wav` jest to $n=9179$. W zależności od długości sygnału ustawić czas symulacji, np. `Simulation Stop time=9000`.
- Przełączyć Manual Switch w położenie From Workspace [`ts,s`] (dwukrotne kliknięcie w obszarze bloku). Przeprowadzić symulację obserwując w oknach wykresów jak parametry filtra adaptują się do zmieniających się w czasie charakterystyk sygnałów wejściowych.
- Odtworzyć na wyjściu audio sygnały obliczone w symulacji i wykreślić przebiegi zaszumionego sygnału użytecznego (sygnał odniesienia `d` filtra kasowania szumu) oraz odtworzonej (po kasowaniu szumu) estymaty sygnału użytecznego (sygnał odniesienia `e` filtra kasowania szumu).
Uwaga: Do odtworzenia drugiego sygnału i wykreślenia wykresów potrzebne jest naciśnięcie dowolnego klawisza w aktywnym oknie komend.

```
t=ScopeData.time;
```

```
d=ScopeData.signals(1,2).values; % syg.odniesienia = syg.oryginalny + szum
e=ScopeData.signals(1,5).values; % syg.błędu po adapt. kasowaniu szumu
disp('Odtworzenie sygnału z szumem')
sound(d,fs)
disp('Press any key...'), pause
disp('Odtworzenie sygnału po adaptacyjnym kasowaniu szumu')
sound(e,fs)
nw=[1,6000]; % zakres czasowy rysowania wykresów
figure
subplot(211), plot(t(nw(1):nw(2)), s(nw(1):nw(2))), grid, ylim([-1,1])
ylabel('s(n)')
title('Adaptacyjne kasowanie szumu: sygnał oryginalny i odtworzony')
subplot(212), plot(t(nw(1):nw(2)), e(nw(1):nw(2)),'r'), grid, ylim([-1,1])
ylabel('e(n) = s_{est}(n)'), xlabel('n')
```

- Ocenic „na ucho” uzyskane efekty kasowania szumu. Czy przy zadanym natężeniu szumu w (zazumionym) sygnale odniesienia słyszalna jest zawartość sygnału użytecznego?
- Z czym związane są – widoczne na wykresie - duże wartości początkowych 100-200 próbek sygnału błędu estymacji?
- Wyznaczyć ilościowy wskaźnik jakości estymacji LMS sygnału użytecznego za pomocą oszacowania stosunku sygnału do szumu błędu estymacji: $SNR = 10 \log \frac{\text{var } s}{\text{var}(s - e)}$ (dB), zastosowanego do zapisanych w pamięci wektorów próbek:

```
stop=9000;
SNRLMS = 10*log10(sum(s(1:stop).^2) / sum((s(1:stop)-e(1:stop)).^2))
```

- E.** Zamienić filtr adaptacyjny w schemacie blokowym na filtr RLS jak w punkcie 3.3.E i powtórzyć eksperyment kasowania szumu w sygnale dźwiękowym jak w pkt. D.

Parametry filtra RLS:

RLS Filter: Filter length=41, Forgetting factor=1.0 (współczynnik zapominania λ), Initial value of weights=0.0, Initial input variance estimate=0.1

- Na podstawie obserwacji procesu adaptacyjnego kasowania w czasie symulacji, wykresów i dźwiękowego odtworzenia estymaty sygnału oryginalnego ocenic uzyskane efekty kasowania szumu za pomocą filtra RLS. Wyznaczyć – jak poprzednio - ilościowy wskaźnik jakości estymacji RLS:

```
stop=9000;
SNRRLS = 10*log10(sum(s(1:stop).^2) / sum((s(1:stop)-e(1:stop)).^2))
```

- Porównać wyniki uzyskane dla filtracji nLMS i RLS.

- (*do uznania prowadzącego ćwiczenia*) Zmienić współczynnik zapominania λ w bloku filtra RLS na Forgetting factor=0.98. Przeprowadzić symulację i porównać wyniki z otrzymanymi dla $\lambda=1$.

- F.** Przywrócić Forgetting factor=1.0, przełączyć Manual Switch w położenie Signal generator, skrócić czas symulacji do stosowanego poprzednio dla sygnału z generatora (Stop time=1500) i przeprowadzić symulację z filtrem RLS z parametrami jak w punktach 3.4 A,B,C.

- Porównać wyniki uzyskane dla filtracji NLMS i RLS.

4. Opracowanie sprawozdania

W sprawozdaniu należy zawrzeć zarejestrowane wyniki eksperymentów przeprowadzonych według instrukcji oraz zadanych przez prowadzącego zadań do realizacji samodzielnej z odpowiednimi opisami oraz wyjaśnieniami opartymi na wiedzy z wykładu i literatury, ze szczególnym uwzględnieniem problemów wskazanych symbolem ➤.

Literatura

1. Poularikas A.D., Ramadan Z.M.: *Adaptive Filtering. Primer with Matlab*, CRC - Taylor & Francis, 2006.
2. Diniz P.S.: *Adaptive Filtering. Algorithms and Practical Implementation*, 4 ed., Springer, 2012.
3. Mitra S.K.: *Digital Signal Processing. A Computer Based Approach*, 4 ed., McGraw-Hill, 2011.
4. Leis J.W.: *Digital Signal Processing Using Matlab for Students and Researchers*, John Wiley & Sons, 2011.
5. Orfanidis S.J.: *Introduction to Signal Processing*, Prentice Hall, 2010.
6. Zieliński T.: *Cyfrowe przetwarzanie sygnałów. Od teorii do zastosowań*, WKiŁ, 2005.
7. Smith S.W.: *Cyfrowe przetwarzanie sygnałów. Praktyczny poradnik dla inżynierów i naukowców*, Wyd. BTC, 2007.

Opracował: Dr inż. Janusz Baran

Częstochowa, 1999-2013