

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ
«ДНІПРОВСЬКА ПОЛІТЕХНІКА»



ФАКУЛЬТЕТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ
Кафедра безпеки інформації та телекомунікацій

В. І. Корнієнко, О. В. Герасіна

СУЧАСНІ МЕТОДИ ОБРОБКИ ІНФОРМАЦІЇ

Методичні рекомендації до виконання практичних робіт
для здобувачів ступеня магістра
освітньо-професійної програми «Телекомунікації та радіотехніка»
зі спеціальності 172 Електронні комунікації та радіотехніка

У 2 частинах

Частина 2

Дніпро
НТУ «ДП»
2024

Корнієнко В.І.

Сучасні методи обробки інформації [Електронний ресурс] : методичні рекомендації до виконання практичних робіт для здобувачів ступеня магістра освітньо-професійної програми «Телекомунікації та радіотехніка» зі спеціальності 172 Електронні комунікації та радіотехніка : у 2 ч. Ч 2 / В. І. Корнієнко, О. В. Герасіна ; М-во освіти і науки України, Нац. техн. ун-т «Дніпровська політехніка». – Дніпро : НТУ «ДП», 2024. – 43 с.

Автори:

В.І. Корнієнко, д-р техн. наук, проф.

О. В. Герасіна, канд. техн. наук, доц.

Затверджено науково-методичною комісією зі спеціальності 172 Електронні комунікації та радіотехніка (протокол № 1 від 29.08.2024) за поданням кафедри безпеки інформації та телекомунікацій (протокол № 1 від 29.08.2024).

Призначено для виконання практичних робіт з дисципліни «Сучасні методи обробки інформації» студентами спеціальності 172 Електронні комунікації та радіотехніка.

Орієнтовано на активізацію навчальної діяльності магістрів та закріплення практичних знань з даної дисципліни.

Відповідальний за випуск завідувач кафедри безпеки інформації та телекомунікацій В.І. Корнієнко, д-р техн. наук, проф.

ЗМІСТ

Вступ	4
Практична робота № 1	
«Нейромережевий амплітудний детектор»	6
Практична робота № 2	
«Прогнозуючий нечіткий фільтр»	15
Практична робота № 3	
«Вейвлет-перетворення векторних сигналів»	24
Практична робота № 4	
«Банки вейвлет-фільтрів»	35
Критерії оцінювання практичних робіт	41
Рекомендовані джерела інформації	42

ВСТУП

Наразі характерним є широке застосування методів систем штучного інтелекту (нейронних мереж, систем нечіткого висновку, еволюційного моделювання тощо) та методів нелінійної динаміки (теорії хаосу та фрактального аналізу) для моделювання складних процесів в системах телекомунікацій та радіотехніки.

Їх актуальність обумовлена, по-перше, спроможністю методів нелінійної динаміки більш узагальнено з єдиних позицій описувати складні процеси в системах різної природи, а, по-друге, здатністю інтелектуальних методів розв'язувати оптимізаційні задачі, що погано формалізуються, та використанням для моделювання ефективних і універсальних апроксиматорів (нейронних мереж та систем нечіткого висновку).

Мета дисципліни – формування у студентів компетентностей щодо володіння сучасними комп'ютерними методами інтелектуальної обробки інформації, методами оптимізації структури, параметрів і характеристик телекомунікаційних та радіотехнічних систем, а також використання фрактальних самоподібних моделей процесів при комп'ютерному проектуванні та аналізі телекомунікаційних систем та мереж, включно об'єктів критичної інфраструктури.

Завдання курсу:

- ознайомити здобувачів вищої освіти із основами технології штучних нейронних мереж та їх ефективністю;
- ознайомити здобувачів вищої освіти з основами нечіткої логіки, алгоритмами побудови систем нечіткого висновку, алгоритмами нечіткої кластеризації; гібридними нейронечіткими мережами, а також із ефективністю усіх вищевказаних систем;
- ознайомити здобувачів вищої освіти із методами еволюційного моделювання (у тому числі із методами групового урахування аргументів);

- ознайомити здобувачів вищої освіти із агентськими алгоритмами оптимізації (мурашиними та бджолиними);
- ознайомити здобувачів вищої освіти із основами нелінійної динаміки та теорії хаосу;
- ознайомити здобувачів вищої освіти з фрактальним аналізом динамічних процесів;
- ознайомити здобувачів вищої освіти з основами частотно-часового аналізу сигналів, породжуваних нелінійними динамічними процесами;
- ознайомити здобувачів вищої освіти з практичними застосуваннями інтелектуальних методів при рішенні завдань в системах телекомунікацій та радіотехніки;
- ознайомити здобувачів вищої освіти із програмним забезпеченням імітаційного моделювання інтелектуальних методів обробки інформації в системах телекомунікацій та радіотехніки.

Результати навчання

Володіти комп'ютерними додатками, що реалізують сучасні методи інтелектуальної обробки інформації в телекомунікаційних та радіотехнічних системах та мережах.

Володіти сучасними інтелектуальними методами оптимізації структури, параметрів і характеристик телекомунікаційних та радіотехнічних систем. Застосовувати нелінійні моделі процесів при інтелектуальній адаптації систем в умовах невизначеності їх стану.

Використовувати сучасні фрактальні самоподібні моделі процесів при проектуванні та аналізі телекомунікаційних систем та мереж, радіорелейних та супутникових систем зв'язку, їх параметрів та характеристик.

ПРАКТИЧНА РОБОТА № 1

«НЕЙРОМЕРЕЖЕВИЙ АМПЛІТУДНИЙ ДЕТЕКТОР»

Мета роботи: дослідження детектування амплітуд часових сигналів за допомогою нейронної мережі.

Програма роботи

1. Попередня підготовка до роботи.
2. Дослідження структури, параметрів і принципу роботи нейромережевого амплітудного детектора.
3. Дослідження ефективності нейромережевого амплітудного детектора.

Етапи роботи

1. Попередня підготовка до роботи

Використовуючи конспект лекцій і літературу засвоїти основи теорії нейронних мереж і принципи побудови амплітудних детекторів.

2. Дослідження структури, параметрів і принципу роботи нейромережевого амплітудного детектора.

Завантажте **Matlab**. Пройдіть шлях **Start/Toolboxes/Neural Network/Demos/Application examples/Elman Amplitude Detection**, виконайте **Run this demo** і ознайомтесь з послідовністю виконуваних етапів у командному рядку.

ЗАДАЧА: розпізнати і виміряти амплітуди синусоїдальних сигналів за допомогою рекурентної нейронної мережі Елмана.

```
p1 = sin(1:20); % Якщо вхідний сигнал – синусоїда з амплітудою 1,  
t1 = ones(1,20); % то на виході амплітудного детектора повинна бути 1  
p2 = sin(1:20)*2; % Якщо вхідний сигнал – синусоїда з амплітудою 2,
```

```
t2 = ones(1,20)*2; % то на виході амплітудного детектора повинне бути 2
```

```
% Навчальна послідовність включає (з чергуванням) дві амплітуди двох  
% синусоїд.
```

```
p = [p1 p2 p1 p2]; t = [t1 t2 t1 t2];
```

```
Pseq = con2seq(p); Tseq = con2seq(t);
```

Формування мережі Елмана з діапазоном зміни входу від -2 до 2 , що має 10 нейронів у схованому шарі і 1 вихід. При цьому функцією активації схованого шару є гіперболічний тангенс (`tansig`), а вихідного шару – лінія (`purelin`). Навчання мережі виконується за допомогою градієнтного алгоритму зворотного поширення (`traingdx`) з обмеженнями по кількості епох (500) і точності навчання (0.01).

```
NN=10; % Кількість нейронів у схованому шарі
```

```
net = newelm([-2 2],[NN 1],{'tansig','purelin'},'traingdx');
```

```
net.trainParam.epochs = 500; net.trainParam.show = 5;
```

```
net.trainParam.goal = 0.01;
```

```
tic % Запуск таймера
```

```
[net,tr] = train(net,Pseq,Tseq); % Навчання мережі
```

```
toc % таймер стоп
```

```
semilogy(tr.epoch,tr.perf)
```

```
% Логарифмічна шкала середньоквадратичної помилки навчання (рис. 1.1).
```

```
title('Mean Squared Error of Elman Network')
```

```
xlabel('Epoch')
```

```
ylabel('Mean Squared Error')
```

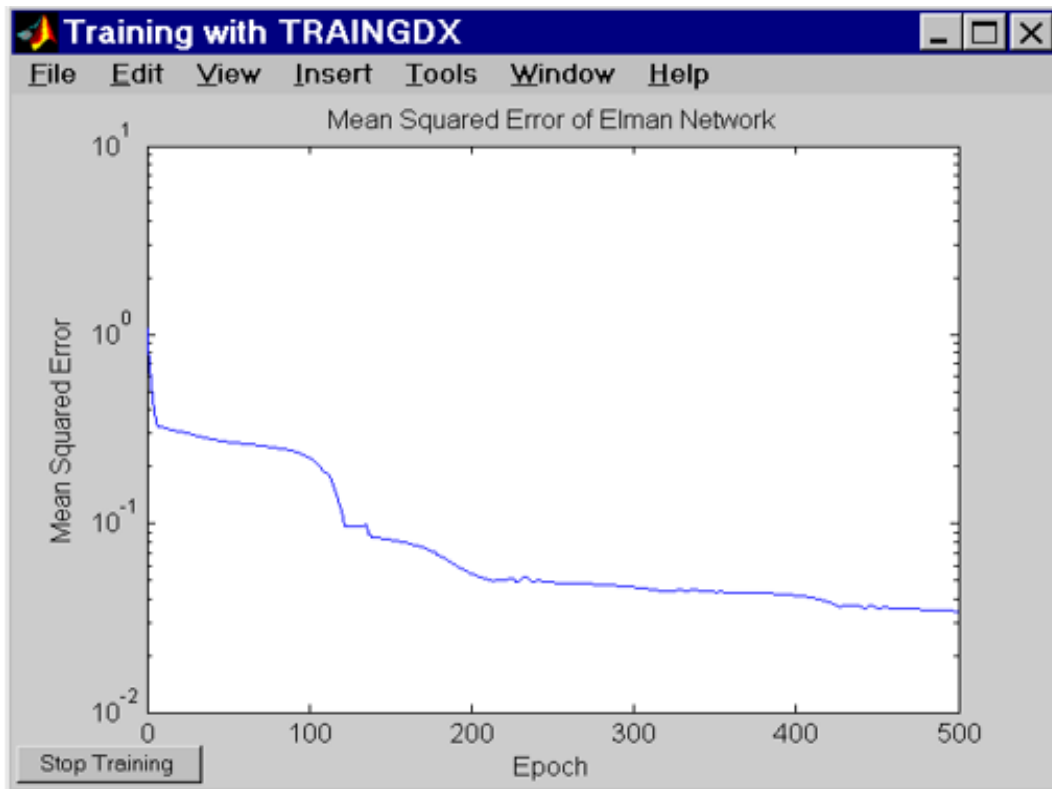


Рис. 1.1. Середньоквадратична помилка навчання нейромережі Елмана

```
a = sim(net,Pseq);
```

```
% Моделювання навченої мережі Елмана при подачі вхідного вектора Pseq
```

```
time = 1:length(p);
```

```
plot(time,t,'--',time,cat(2,a{:}))
```

```
title('Testing Amplitude Detection')
```

```
xlabel('Time Step')
```

```
ylabel('Target - - Output ---')
```

Вигляд виходу нейромережевого амплітудного детектора (зелений) і значення амплітуд синусоїдальних вхідних сигналів (синій) навчальної послідовності наведені на рис. 1.2.

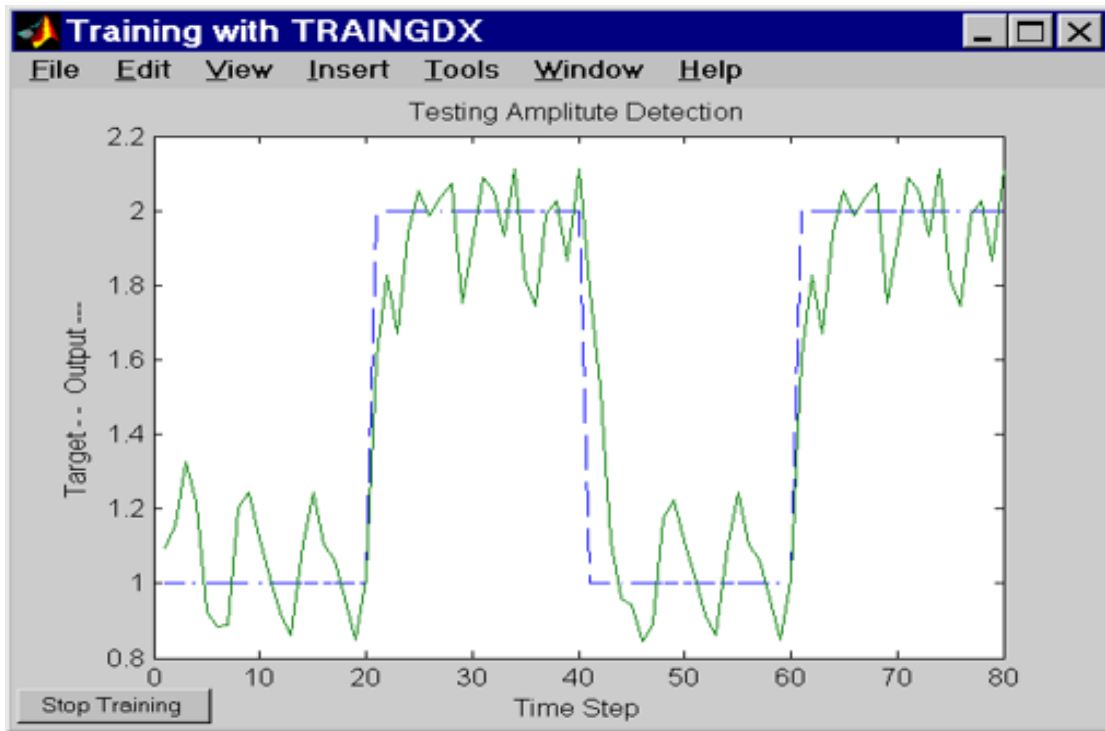


Рис. 1.2. Результати роботи детектора на навчальній послідовності

```
e1=t-cat(2,a{:}); ea1=sum(abs(e1))/mean(t)/length(p)
```

```
% Відносна помилка навчання
```

Для оцінки якості роботи амплітудного детектора перевіримо його на перевіірчній послідовності синусоїдальних сигналів з амплітудами, відмінними від навчальної послідовності.

```
** AV=1.6; AN=1.2; % Амплітуди перевіірочної послідовності
```

```
p3 = sin(1:20)*AV; % Якщо вхідний сигнал - синусоїда з амплітудою AV,
```

```
t3 =ones(1,20)*AV; % то на виході детектора повинне бути AV.
```

```
p4 = sin(1:20)*AN; % Якщо вхідний сигнал - синусоїда з амплітудою AN,
```

```
t4 = ones(1,20)*AN; % то на виході детектора повинне бути AN.
```

```
% Перевіірчна послідовність також включає (з чергуванням) дві амплітуди
```

```
% двох синусоїд.
```

```
pg = [p3 p4 p3 p4]; tg = [t3 t4 t3 t4]; pgseq = con2seq(pg);
```

```
a = sim(net,pgseq);
```

% детектування навченою мережею Елмана вхідного вектора **pg**

```
time = 1:length(pg);
```

```
plot(time,tg,'--',time,cat(2,a{:}))
```

```
title('Testing Generalization')
```

```
xlabel('Time Step')
```

```
ylabel('Target - - Output ---')
```

Вигляд виходу нейромережевого амплітудного детектора (зелений) і значення амплітуд синусоїдальних вхідних сигналів (синій) перевірконої послідовності наведені на рис. 1.3.

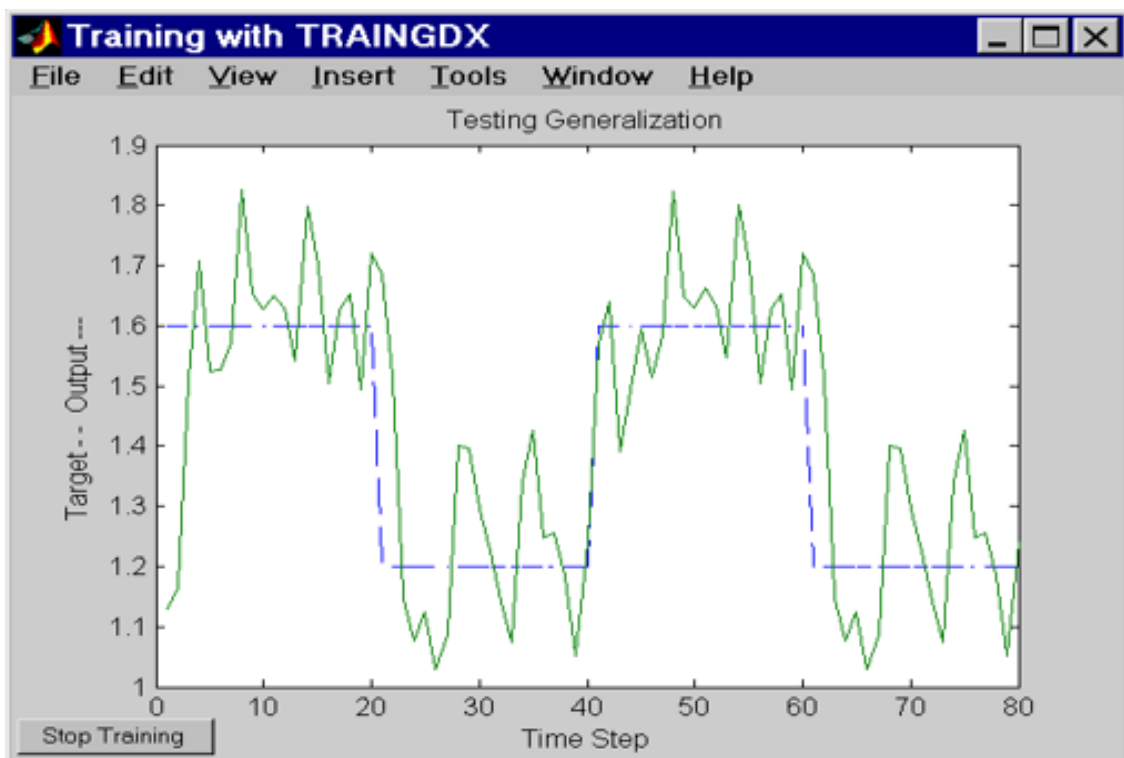


Рис. 1.3. Результати роботи детектора на перевірочній послідовності

$e2 = \text{tg-cat}(2, a{:}); ea2 = \text{sum}(\text{abs}(e2)) / \text{mean}(\text{tg}) / \text{length}(\text{pg})$

% Відносна помилка перевірки

Отримані результати показують, що мережа розпізнає амплітуди, для яких не було навчання.

Виконувати дослідження структури нейронної мережі і її роботу зручно за допомогою імітаційної моделі Simulink (рис. 1.4).

`gensim(net,1)` % Загальна структура моделі.

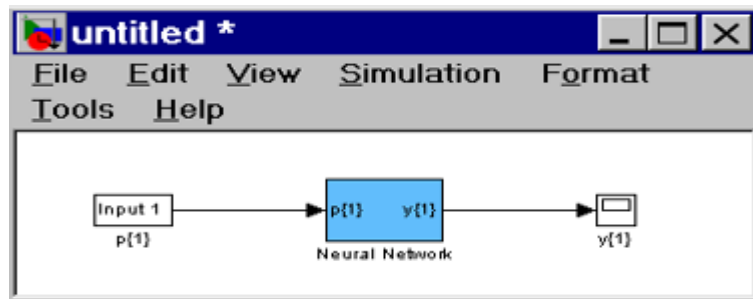


Рис. 1.4. Загальна структура моделі нейронної мережі Елмана

Дворівнева модель рекурентної мережі Елмана представлена на рис. 1.5.

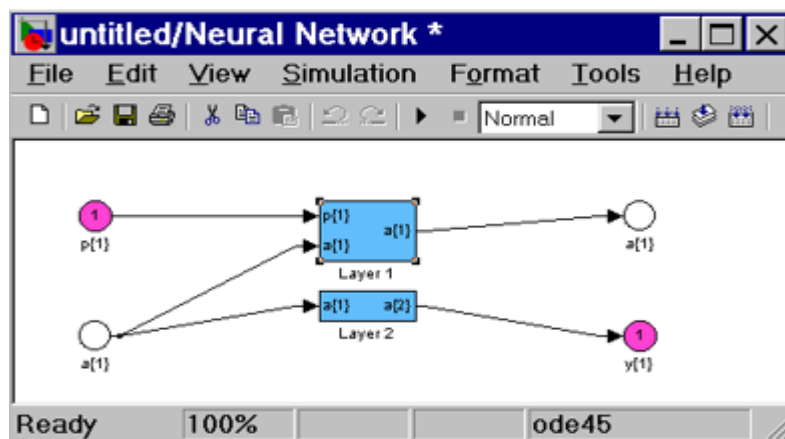


Рис. 1.5. Дворівнева модель нейромережі Елмана

Структура першого шару мережі Елмана представлена на рис. 1.6.

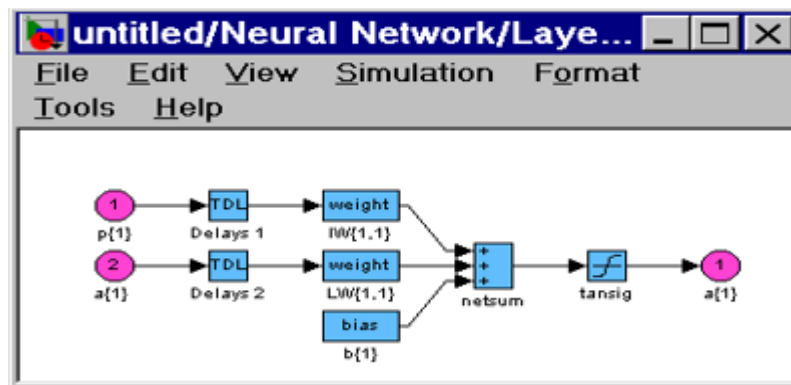


Рис. 1.6. Структура першого шару нейронної мережі Елмана

Схований шар нейронів мережі Елмана представлено на рис. 1.7.

Структура другого шару мережі Елмана представлена на рис. 1.8.

Розглянута мережа Елмана після навчання досить задовільно детектує сигнали з невідомими їй (по навчанню) амплітудами. Кращих результатів можна досягти при збільшенні: навчальної послідовності, кількості епох навчання і нейронів у схованому шарі. Крім того, деякого ефекту можна досягти варіюючи функціями активації й алгоритмом навчання мережі.

3. Дослідження ефективності нейромережевого амплітудного детектора.

3.1. Вставивши в m-файл свої вихідні дані (замінити оператори, позначені **, на наведені нижче оператори) і послідовно копіюючи програму в командний рядок, виконати моделювання.

** $AV=1.7+PN/20$; $AN=1+PN/20$; % PN - Порядковий номер студента

3.2. Зафіксувати скріншоти виходу мережі й отриманих значень відносної помилки амплітудного детектування по навчальній (e1) і перевіірочній (e2) послідовностям сигналів.

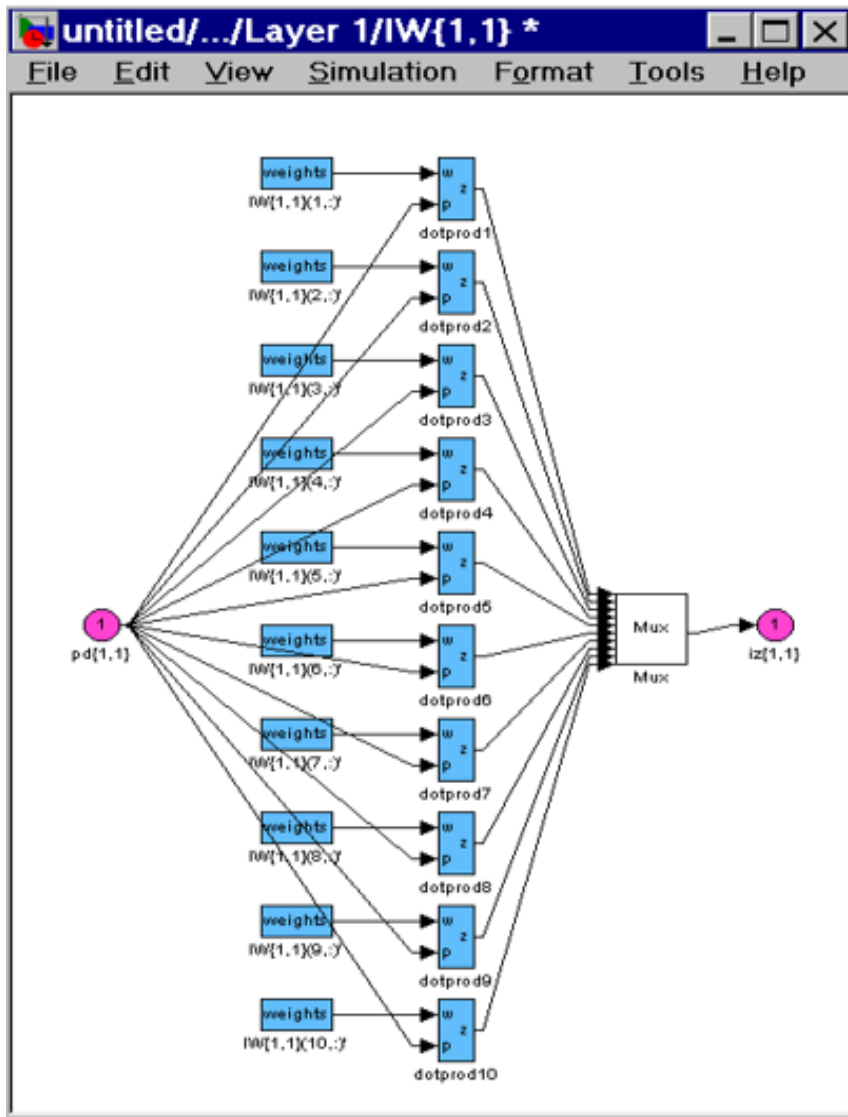


Рис. 1.7. Структура схованого шару нейронної мережі Елмана

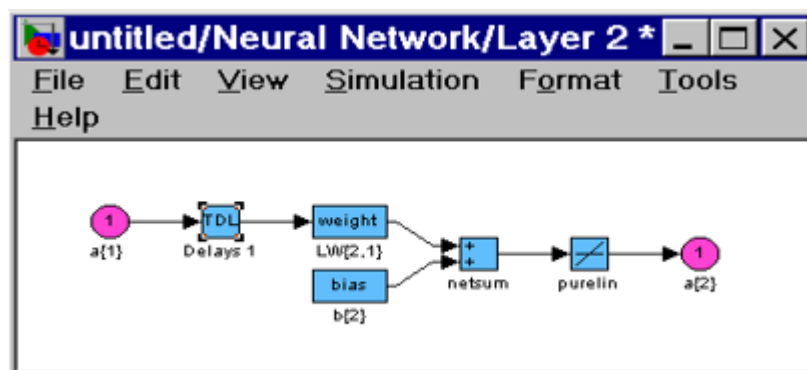


Рис. 1.8. Структура другого шару нейронної мережі Елмана

3.3. Визначити шляхом моделювання і побудувати графіки залежності відносних помилок детектування e_1 , e_2 при варіації кількості нейронів у схованому шарі (NN) від 5 до 50.

Примітка. Моделювання з новою кількістю нейронів проводиться в два етапи:

- 1) настроювання мережі по навчальній послідовності;
- 2) оцінка помилки детектування по перевіірочній послідовності сигналів.

Вимоги до оформлення звіту

Звіт повинний містити:

1. Найменування і мету роботи.
2. Скріншоти виходу амплітудного детектора при варіації параметрів мережі.
3. Графіки залежності відносних помилок детектування від варіації кількості нейронів у схованому шарі з висновками по ним.

Контрольні питання

1. Що таке амплітудне детектування?
2. Яка структура нейронної мережі Елмана?
3. Які області застосування мають нейронні мережі?
4. Які шляхи зниження помилки нейромережевого амплітудного детектування?

ПРАКТИЧНА РОБОТА № 2

«ПРОГНОЗУЮЧИЙ НЕЧІТКИЙ ФІЛЬТР»

Мета роботи: дослідження прогнозування хаотичного процесу з шумом за допомогою нечіткого фільтра.

Програма роботи

1. Попередня підготовка до роботи.
2. Дослідження структури, параметрів і принципу роботи прогнозуючої системи адаптивної нечіткої фільтрації.
3. Дослідження структури і параметрів програмної реалізації прогнозуючої системи адаптивної нечіткої фільтрації.
4. Дослідження ефективності прогнозуючої нечіткої фільтрації.

Етапи роботи

1. Попередня підготовка до роботи

Використовуючи конспект лекцій і літературу засвоїти основи теорії фільтрації, нечіткої логіки і принципи побудови адаптивних систем нечіткого висновку.

2. Дослідження структури, параметрів і принципу роботи прогнозуючої системи адаптивної нечіткої фільтрації.

2.1. Завантажте **Matlab**. Пройдіть шлях **Start/Toolboxes/Fuzzy Logic/Demos/Chaotic Time-Series Prediction**, виконайте **Run this demo**. Або натисніть **F1**, і у вікні **Help** оберіть **Fuzzy Logic Toolbox**, далі **Examples**, і там знайдіть приклад **Chaotic Time-Series Prediction**, виконайте **Open this Example**. Перед Вами відкриється інтерфейс (рис. 2.1).

Даний інтерфейс дозволяє прогнозувати хаотичний процес з шумом за допомогою адаптивної нейронної системи нечіткого висновку (ANFIS).

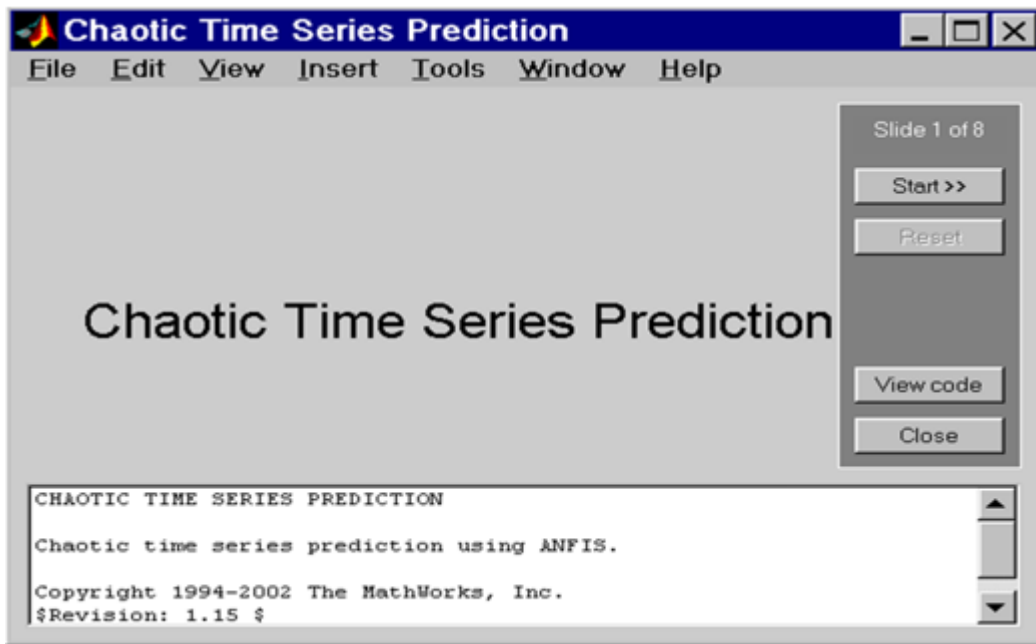


Рис. 2.1. Імітаційна модель прогнозування хаотичного процесу за допомогою ANFIS

Хаотичний процес описується диференціальним рівнянням із запізнюванням Макей-Гласа (Mackey-Glass): $dx(t)/dt = 0.2x(t-\tau)/(1+x(t-\tau)^{10}) - 0.1x(t)$. Коли $x(0) = 1.2$ і $\tau = 17$, то маємо неперіодичний, що не сходиться в часі, процес, який дуже чутливий до початкових умов. (Приймаємо $x(t) = 0$ для $t < 0$). Крім того, додаємо в цей процес незалежний шум (наприклад, шум вимірів) **Noise**, рівний **0...1** від рівня сигналу.

2.2. Запустіть (кнопка “Start”) демонстраційну модель і ознайомтесь з послідовністю виконуваних етапів.

3. Дослідження структури і параметрів програмної реалізації прогнозуючої системи адаптивної нечіткої фільтрації.

За допомогою кнопки «View Code» відкрийте лістинг програмної реалізації системи (представлена у вигляді m-файлу) і вивчіть її.

** Noise=0.0; % Рівень шуму

**** PN=0; % Порядковий номер студента в списку групи**

```
load mgdata.dat
```

```
a=mgdata; time = a(:, 1); ts1 =0.9*a(:, 2)+PN/50;
```

```
ts2=ts1+Noise*rand(size(ts1)); % Сигнал+Шум
```

```
ts=ts2-(mean(ts2)-mean(ts1));
```

```
plot(time, ts1); xlabel('Time (sec)'); ylabel('x(t)'); % Графік Сигналу
```

```
title('Mackey-Glass Chaotic Time Series');
```

```
figure;
```

```
plot(time, ts); xlabel('Time (sec)'); ylabel('x(t)');
```

```
title('Mackey-Glass Chaotic Time Series+Noise'); % Графік Сигнал+Шум
```

На рис. 2.2 представлено графік, що показує відрізок часу використуваних даних.

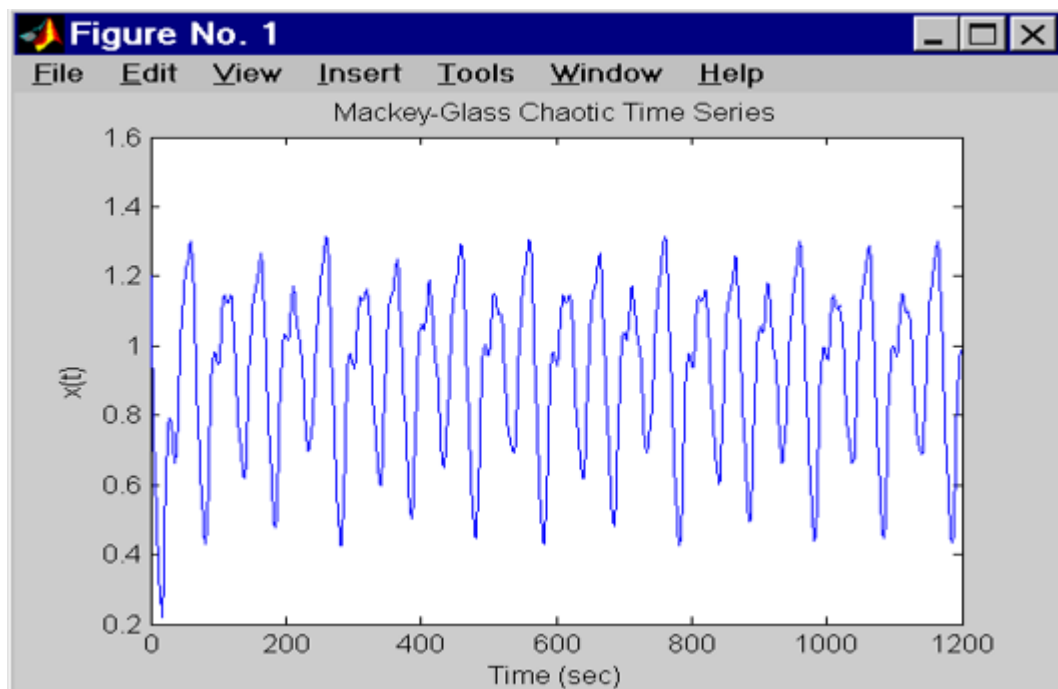


Рис. 2.2. Вхідні дані для моделювання

Необхідно побудувати нечіткий фільтр у вигляді адаптивної системи ANFIS, що прогнозує значення процесу $x(t+6)$ по серії його минулих значень, тобто: $x(t-18)$, $x(t-12)$, $x(t-6)$, і $x(t)$. Отже, формат вихідних даних - $[x(t-18), x(t-12), x(t-6), x(t); x(t+6)]$. З $t = 118$ до 1117 вибираємо **1000** даних вищевказаного формату, перші **500** з яких використовуються для навчання фільтра, а другі являють собою перевірочну послідовність.

```
trn_data = zeros(500, 5); chk_data = zeros(500, 5);
```

```
% підготовка навчальної послідовності
```

```
start = 101; trn_data(:, 1) = ts(start:start+500-1);
```

```
start = start + 6; trn_data(:, 2) = ts(start:start+500-1);
```

```
start = start + 6; trn_data(:, 3) = ts(start:start+500-1);
```

```
start = start + 6; trn_data(:, 4) = ts(start:start+500-1);
```

```
start = start + 6; trn_data(:, 5) = ts(start:start+500-1);
```

```
% підготовка перевірочної послідовності
```

```
start = 601; chk_data(:, 1) = ts(start:start+500-1);
```

```
start = start + 6; chk_data(:, 2) = ts(start:start+500-1);
```

```
start = start + 6; chk_data(:, 3) = ts(start:start+500-1);
```

```
start = start + 6; chk_data(:, 4) = ts(start:start+500-1);
```

```
start = start + 6; chk_data(:, 5) = ts(start:start+500-1);
```

```
index = 118:1117+1; % старт з t = 0
```

```
plot(time(index), ts(index));
```

```
axis([min(time(index)) max(time(index)) min(ts(index)) max(ts(index))]);
```

```
xlabel('Time (sec)'); ylabel('x(t)'); title('Mackey-Glass Chaotic Time Series');
```

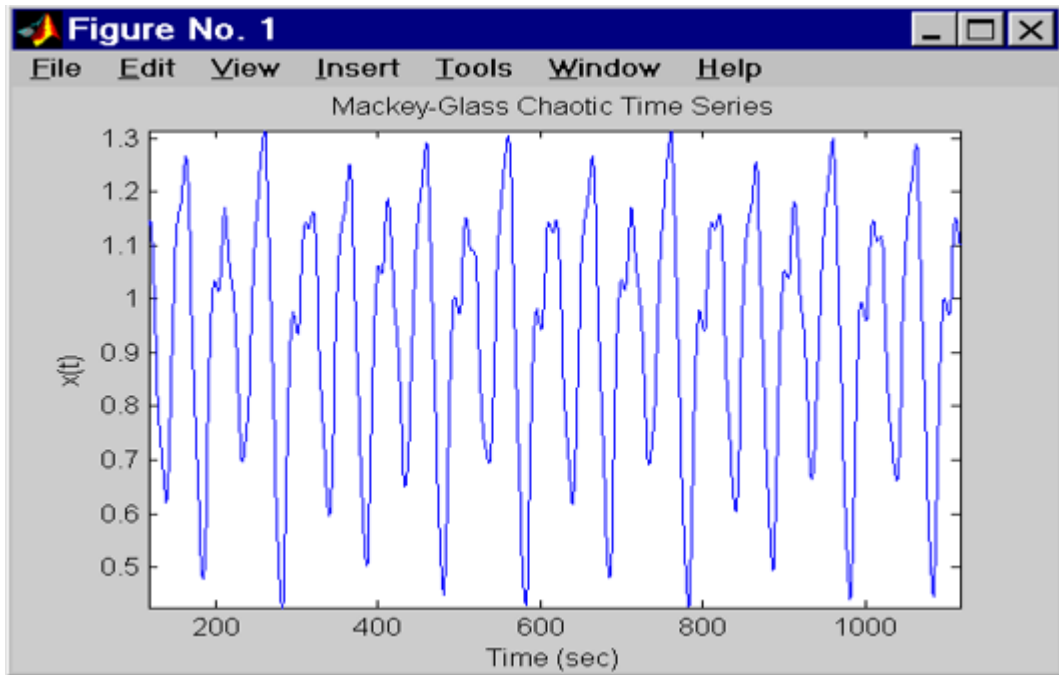


Рис. 2.3. Перевірочна послідовність

Ми використовуємо оператор GENFIS1 для генерування початкової матриці ініціалізації FIS з навчальною послідовністю. Використовується спрощений формат запису цього оператора зі значеннями для кількості нечітких правил (за замовчуванням 2) і типу функції належності MF (за замовчуванням використовується дзвіноподібна функція gbellmf).

```
fismat = genfis1(trn_data);
```

```
% Повна форма = genfis1(trn_data,[3 7],char('pimf','trimf'))
```

% Графіки ініціалізованих MFs для навчання наводяться нижче.

```
for input_index=1:4,  
    subplot(2,2,input_index)  
    [x,y]=plotmf(fismat,'input',input_index);  
    plot(x,y)
```

```
axis([-inf inf 0 1.2]); xlabel(['Input ' int2str(input_index)]);
end
```

figure; plotfis(fismat); % Структура нечіткого фільтра (рис. 2.4)

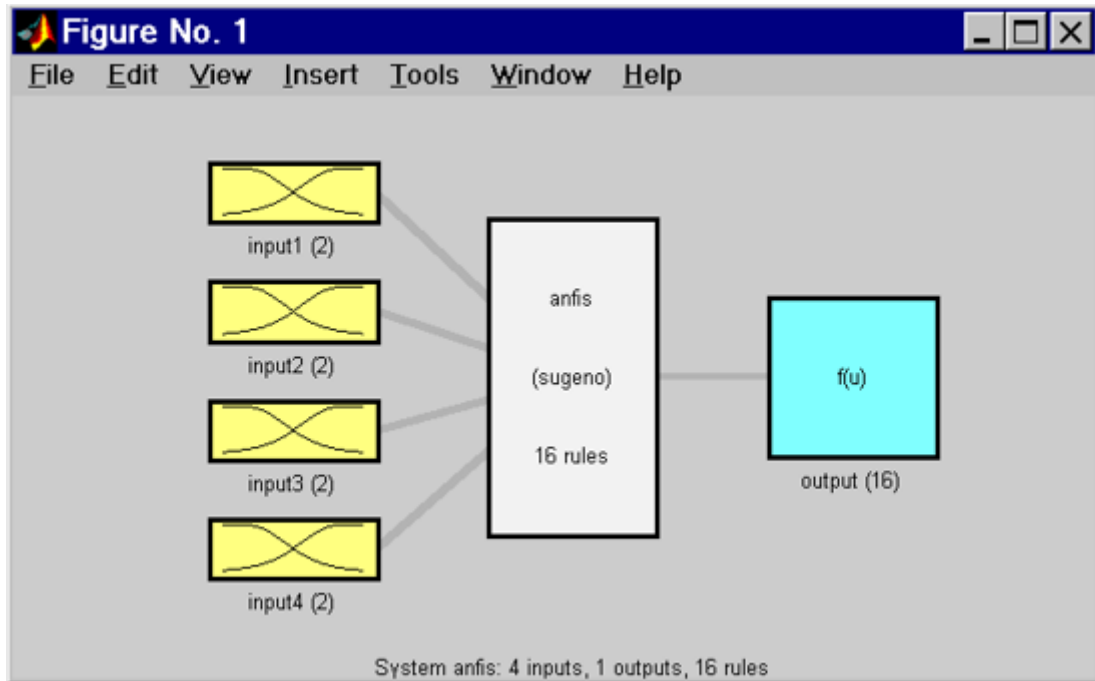


Рис. 2.4. Структура нечіткого фільтра на базі ANFIS

Система ANFIS має $2^4 = 16$ правил у генерованій матриці FIS і кількість параметрів - 108, включаючи 24 нелінійних і 80 лінійних параметрів.

```
Epoch(1)=10; % Кількість епох навчання
[trn fismat, trn_error, KK, KM, chk_error] = anfis(trn_data,
fismat, Epoch(1), [], chk_data)
for input_index=1:4,
    subplot(2,2,input_index)
    [x,y]=plotmf(trn_fismat,'input',input_index); plot(x,y)
    axis([-inf inf 0 1.2]); xlabel(['Input ' int2str(input_index)]);
end
```

Вигляд функцій належності після навчання представлено на рис. 2.5.

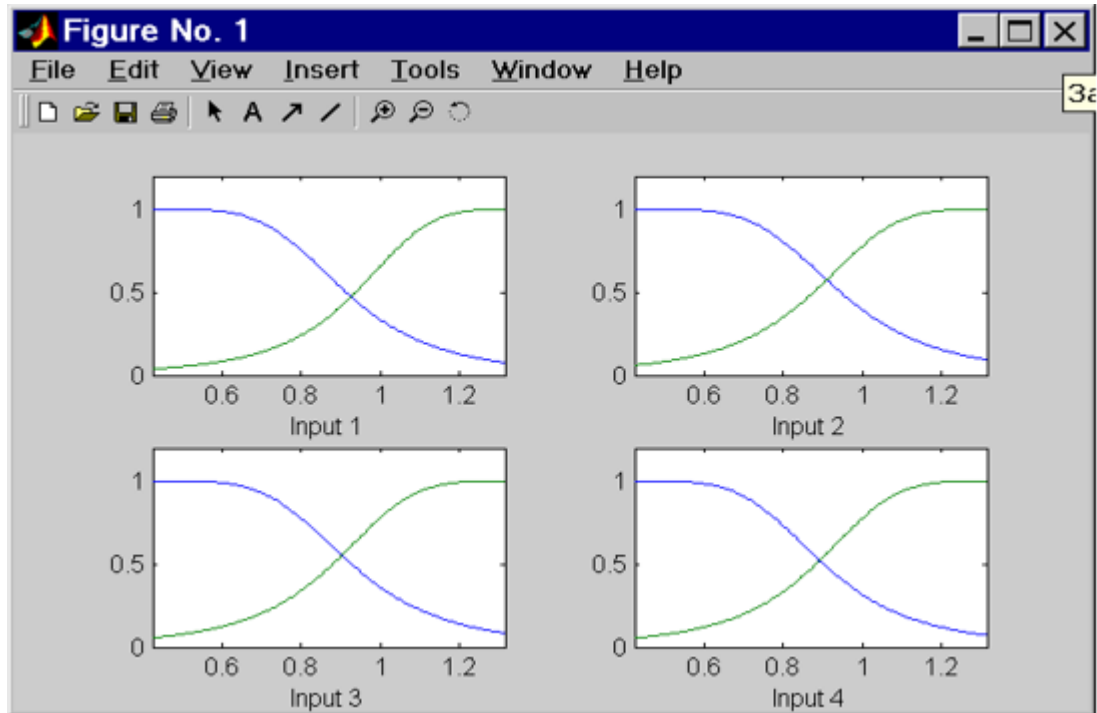


Рис. 2.5. Вигляд функцій належності після навчання

```
input = [trn_data(:, 1:4); chk_data(:, 1:4)]; anfis_output = evalfis(input,  
trn_fismat);  
index = 125:1124; plot(time(index), [ts1(index) anfis_output]); xlabel('Time  
(sec)');
```

На рис. 2.6 наведені графіки вихідного процесу (синій) і виходу прогнозуючого нечіткого фільтра (зелений). При малій помилці прогнозу ці графіки зливаються.

```
diff = ts1(index)-anfis_output; plot(time(index), diff);  
xlabel('Time (sec)'); title('ANFIS Prediction Errors'); d=diff;  
ed=sum(abs(d))/1000/mean(ts1) % Відносна помилка прогнозу
```

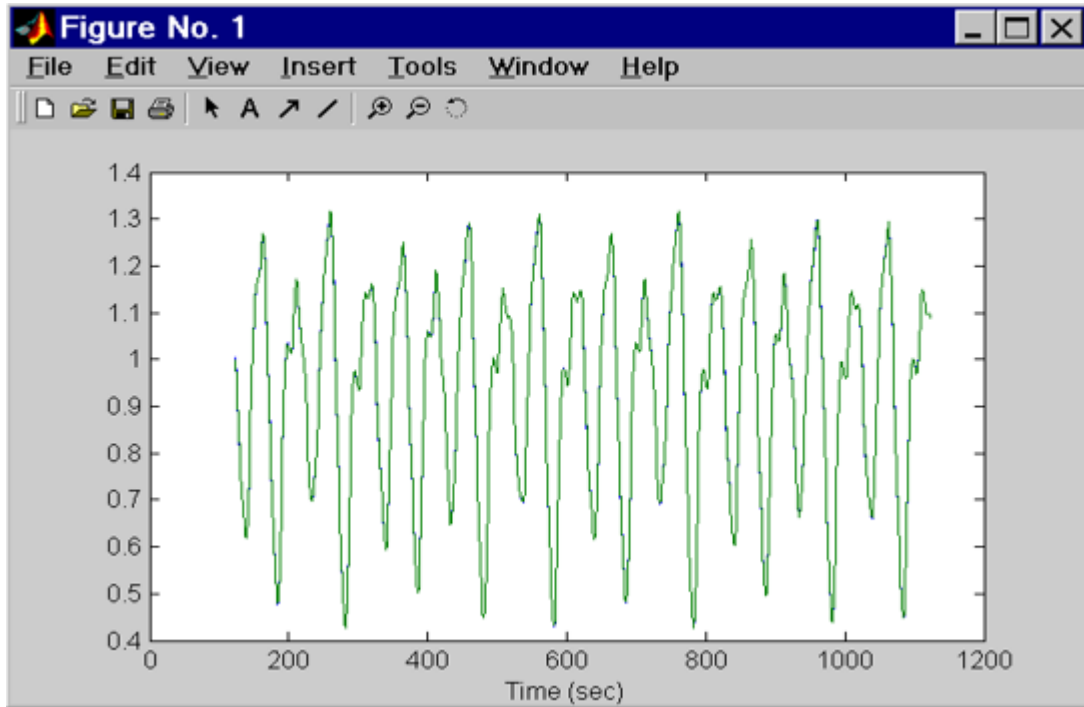


Рис. 2.6. Результат прогнозування

Графік помилки прогнозу показаний на рис. 2.7.

4. Дослідження ефективності прогнозуючої нечіткої фільтрації.

4.1. Вставивши в **m-файл** свої вихідні дані (замінити оператори, позначені ******, на свої дані) і послідовно копіюючи програму в командний рядок, виконати моделювання.

4.2. Зафіксувати скріншоти вхідного процесу і виходу прогнозуючого фільтра, а також величину відносної помилки прогнозу **ed**.

4.3. Визначити шляхом моделювання і побудувати графік залежності відносної помилки прогнозу **ed** від рівня шуму процесу **Noise=0...1**.

4.4. Запропонувати шляхи зниження величини помилки прогнозу й оцінити їх ефективність за допомогою моделювання.

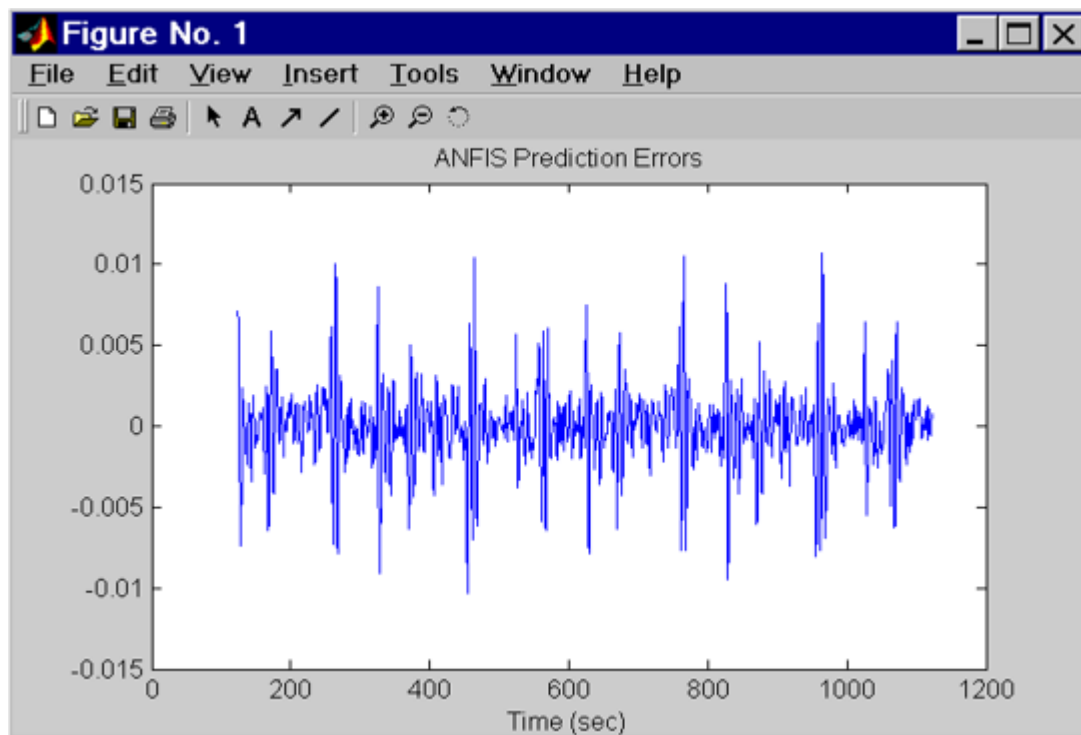


Рис. 2.7. Похибка прогнозування

Вимоги до оформлення звіту

Звіт повинний містити:

1. Найменування і мету роботи.
2. Скріншоти вихідного процесу і виходу прогнозуючого нечіткого фільтра.
3. Графіки залежності відносної помилки прогнозу від рівня шуму процесу з висновками по ним.
4. Результати дослідження шляхів зниження величини помилки прогнозу з висновками по ним.

Контрольні питання

1. Що таке прогнозуючий фільтр?
2. Що таке адаптивна система ANFIS?
3. Які шляхи зниження помилки прогнозу нечіткого фільтра?
4. Які області застосування мають методи нечіткої логіки?

ПРАКТИЧНА РОБОТА № 3

«ВЕЙВЛЕТ-ПЕРЕТВОРЕННЯ ВЕКТОРНИХ СИГНАЛІВ»

Мета роботи: дослідження аналізу і синтезу векторних сигналів, їхнього стиску, а також фільтрації шуму в них за допомогою вейвлет-перетворення.

Програма роботи

1. Попередня підготовка до роботи.
2. Дослідження структури і параметрів пакета програм вейвлет-перетворення векторних сигналів.
3. Дослідження вейвлет-перетворення векторних сигналів.

Етапи роботи

1. Попередня підготовка до роботи

Використовуючи конспект лекцій і літературу засвоїти основи теорії вейвлет-перетворення сигналів.

2. Дослідження структури і параметрів пакета програм вейвлет-перетворення векторних сигналів.

2.1. Пройдіть шлях **Start/Toolboxes/Wavelet/Demos/ Wavelet 1-D**, виконайте **Run this demo** і перед Вами відкриється інтерфейс (рис. 3.1).

Пакет програм (ПП) вейвлет-перетворення (ВВП) векторних (1-D) сигналів забезпечує обробку локальних, багатосмугових і нестационарних сигналів. Методи ВВП пропонують додаткові можливості по обробці сигналів у додатках, де використовується перетворення Фур'є. ВВП ефективно при обробці сигналів і зображень у різних додатках, включаючи мовну й аудіообробку, зв'язок, геофізику, фінанси, медицину тощо. ПП ВВП призначений для аналізу, синтезу, фільтрації і стиску сигналів і зображень.



Рис. 3.1. Імітаційна модель вейвлет-перетворення векторних сигналів

2.2. Запустіть демонстраційну модель і перед Вами відкриються слайди.

Слайд 2

Дана презентація показує можливості ВВП при перетворенні векторних сигналів. У цьому прикладі нестационарний сигнал з шумом Doppler розкладається на 5 рівнів з використанням вейвлета Symlet 4 (sym4).

Слайд 3

Декомпозиція сигналу s являє собою суму його грубої апроксимації (a_5) і деталізації на п'ятьох рівнях ($d_1... d_5$), тобто $s = a_5 + d_5 + d_4 + d_3 + d_2 + d_1$. Оскільки sym4 – ортогональний вейвлет, то і розкладання сигналу – ортогональне (рис. 3.2).

При фільтрації шуму чи стиску сигналу тип вейвлета є настроюваним параметром. Найчастіше вибір вейвлета не критичний і будь-який вейвлет (за винятком Хаара) дає гарне розкладання.

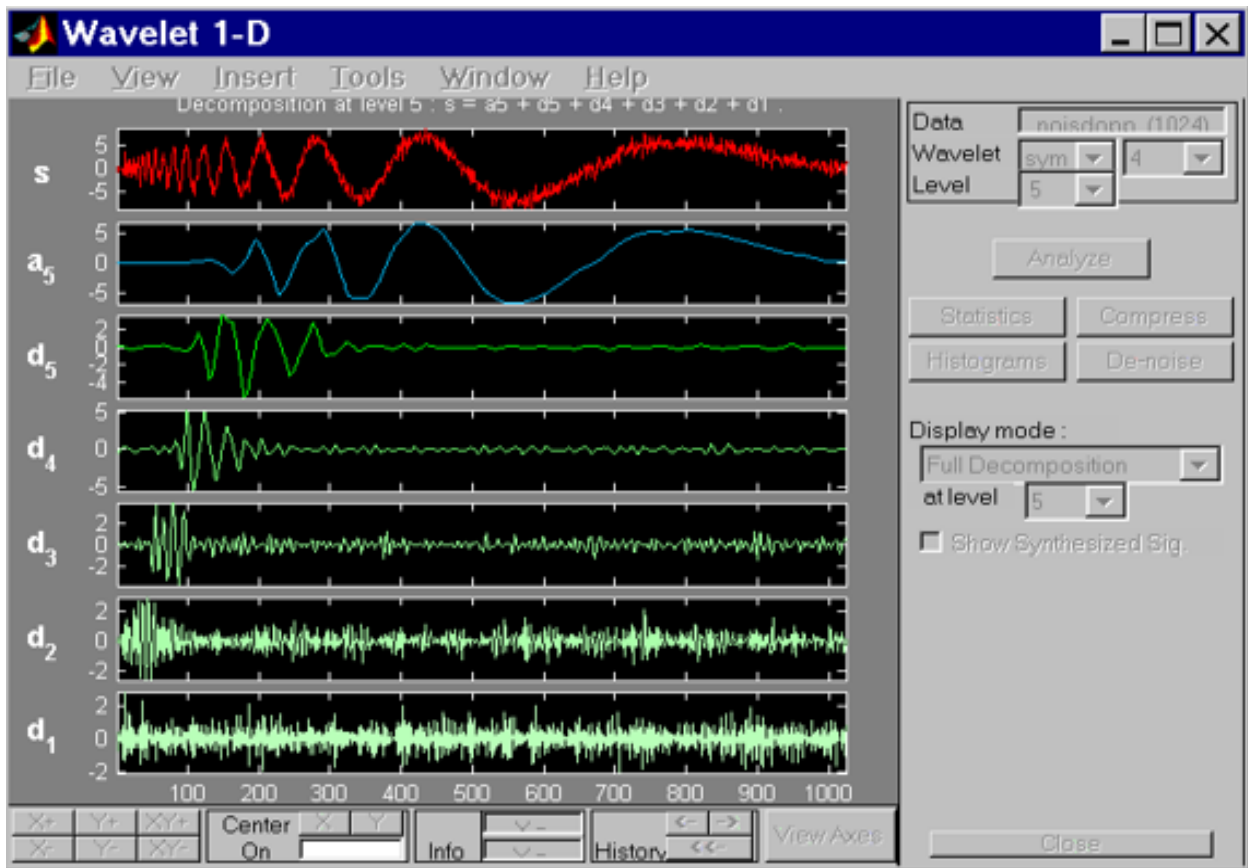


Рис. 3.2. Слайд 3 моделі вейвлет-перетворення векторних сигналів

Вибравши вейвлет, варто встановити кількість рівнів розкладання (декомпозиції).

Слайд 4

Кількість рівнів декомпозиції обирають таким чином, щоб відповідна апроксимація не мала шуму. У цьому прикладі 4 чи 5, - гарні кандидати, у той час, як апроксимації на рівнях 1 - 3 явно мають шум (рис. 3.3).

Графічний інтерфейс користувача ПП ВВП забезпечує гнучкий вибір типу вейвлет, кількості рівнів розкладання, параметрів графічного представлення тощо.

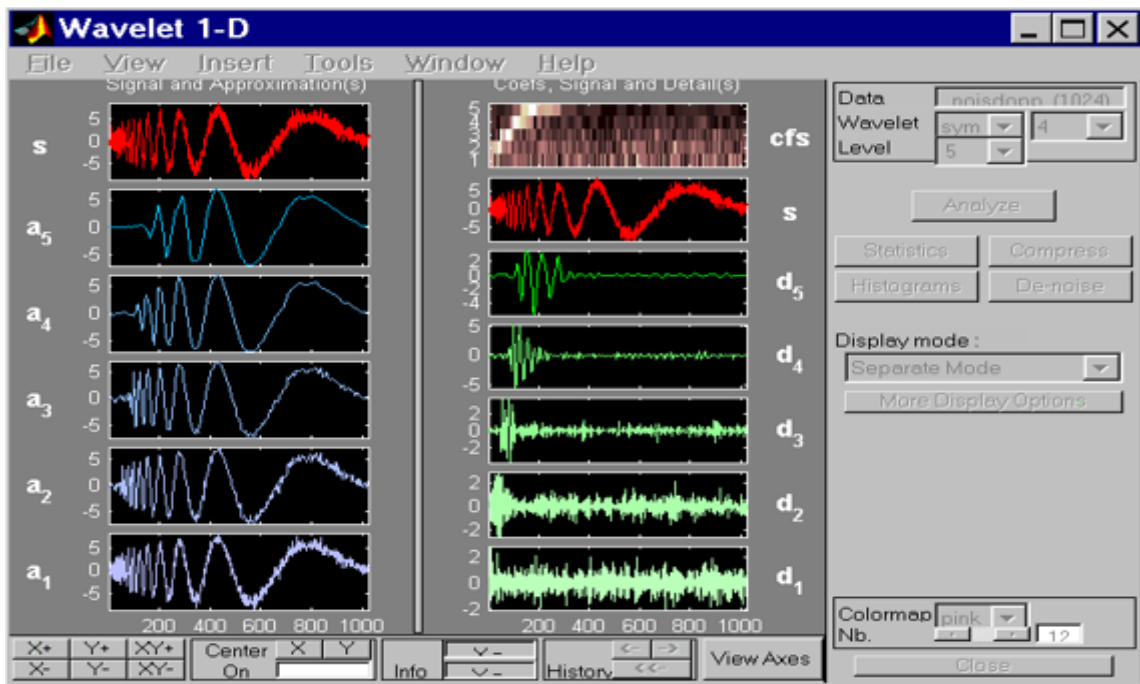


Рис. 3.3. Слайд 4 моделі вейвлет-перетворення векторних сигналів

Слайд 5

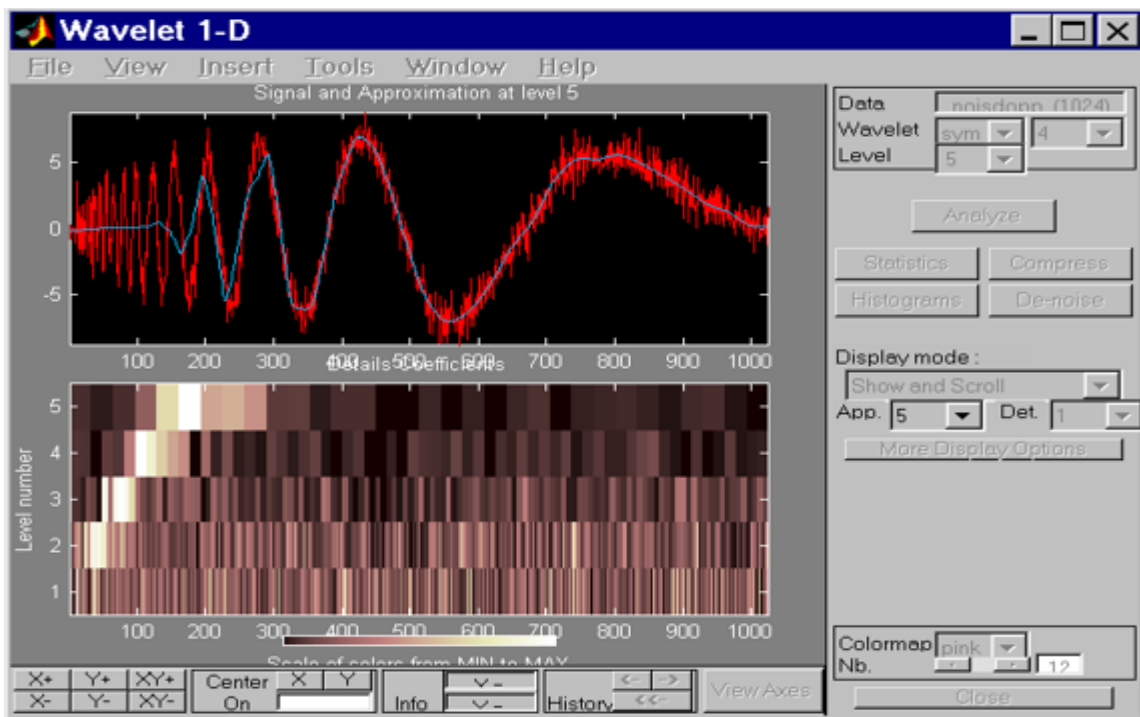


Рис. 3.4. Слайд 5 моделі вейвлет-перетворення векторних сигналів

Слайд 6

Часо-частотне (часо-масштабне) представлення показує, що детермінована частина сигналу сконцентрована в декількох великих коефіцієнтах, а шум розташований у всьому часо-масштабному просторі.

Слайд 7

Клацаючи на кнопку Гістограм і обираючи рівні розкладання, ми можемо бачити, як їх коефіцієнти деталізації апроксимуються нормальним гауссівським розподілом.

Слайд 8

Фільтрація шуму чи процедура стиску полягають у зміні коефіцієнтів деталізації, наприклад, шляхом обнуління найменших коефіцієнтів (рис. 3.5).

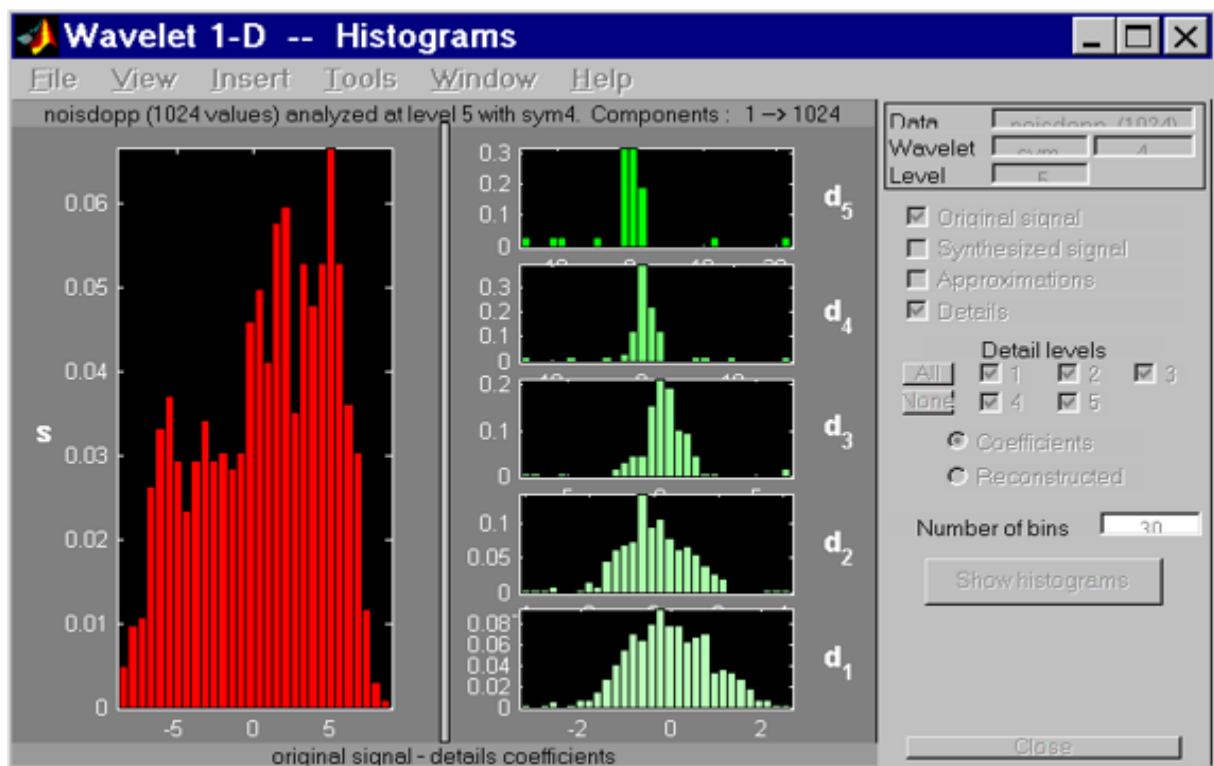


Рис. 3.5. Слайд 8 моделі вейвлет-перетворення векторних сигналів

Слайд 9

Стиск. При натисканні кнопки Стиск з'являється відповідне вікно, що дозволяє споживачам обирати розмір порога (трешолд) для двох параметрів:

Слайд 10

- енергії сигналу, що зберігається (у відсотках), рівна відношенню квадратів стиснутого й оригінального (пурпурна лінія ліворуч) сигналів;
- кількості нульових коефіцієнтів, рівних відсотку нулів представлення стиснутого сигналу (синя лінія ліворуч).

Обидва параметри дають приблизно однакові результати. Пороги встановлюються або на правій числовій панелі, або ліворуч шляхом переміщення мишею глобального порога (вертикальна лінія). Відновлений стиснутий сигнал має більш гладкий вигляд і містить мало коефіцієнтів у часо-масштабному представленні.

Слайд 11

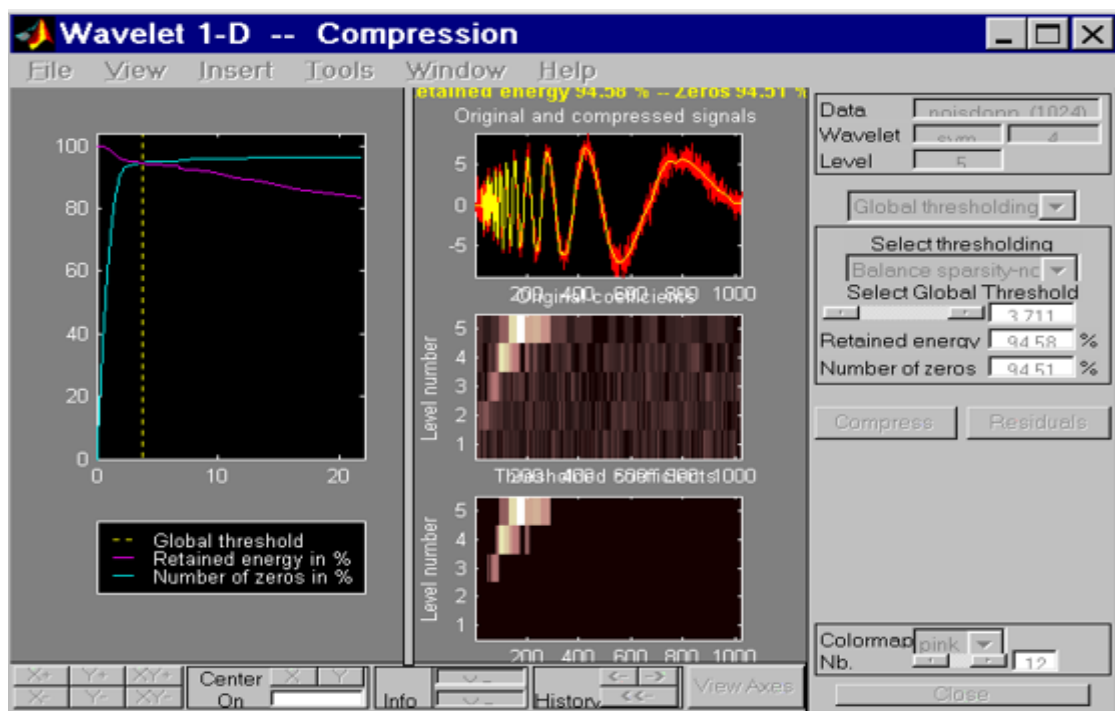


Рис. 3.6. Слайд 11 моделі вейвлет-перетворення векторних сигналів

Слайд 12

Модифікація кожного з трьох параметрів: розмір глобального порога, енергії, що зберігається, чи кількості нулів, - автоматично коректує інші два.

Слайд 13

Щоб довідатися більше про процедуру стиску, виберіть спосіб багаторівневого порога (By-Level thresholding). Пороги коефіцієнтів деталізації представлені жовтими лініями ліворуч. У порівнянні з попереднім способом це дозволяє встановлювати розмір порога для кожного рівня розкладання, що приводить до додаткової гнучкості.

Слайд 14

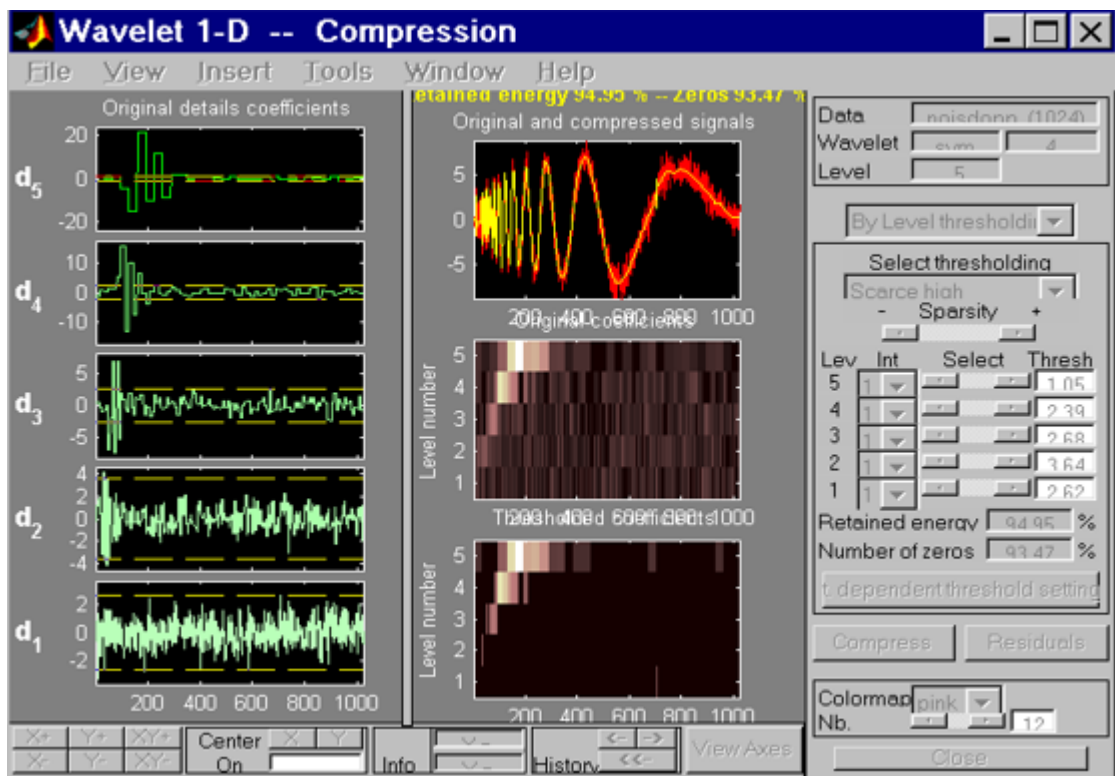


Рис. 3.7. Слайд 14 моделі вейвлет-перетворення векторних сигналів

Слайд 15

Фільтрація шуму. Метою обробки сигналу є його відновлення з видаленням шуму, незалежного від сигналу (шум вважається невідомим). Також вважається, що шум адитивний і стаціонарний (принаймні на кожній шкалі розкладання).

Основні ідеї для фільтрації шуму і стиску однакові. Розходження полягають в наявності додаткових режимів:

- а) використанні м'якого і жорсткого порога (трешолдинга);
- б) вибору структури аналізованого шуму (масштабований чи ні білий шум, кольоровий шум).

Засоби графічного інтерфейсу користувача для фільтрації шуму подібні засобам режиму Стиску.

Слайд 16

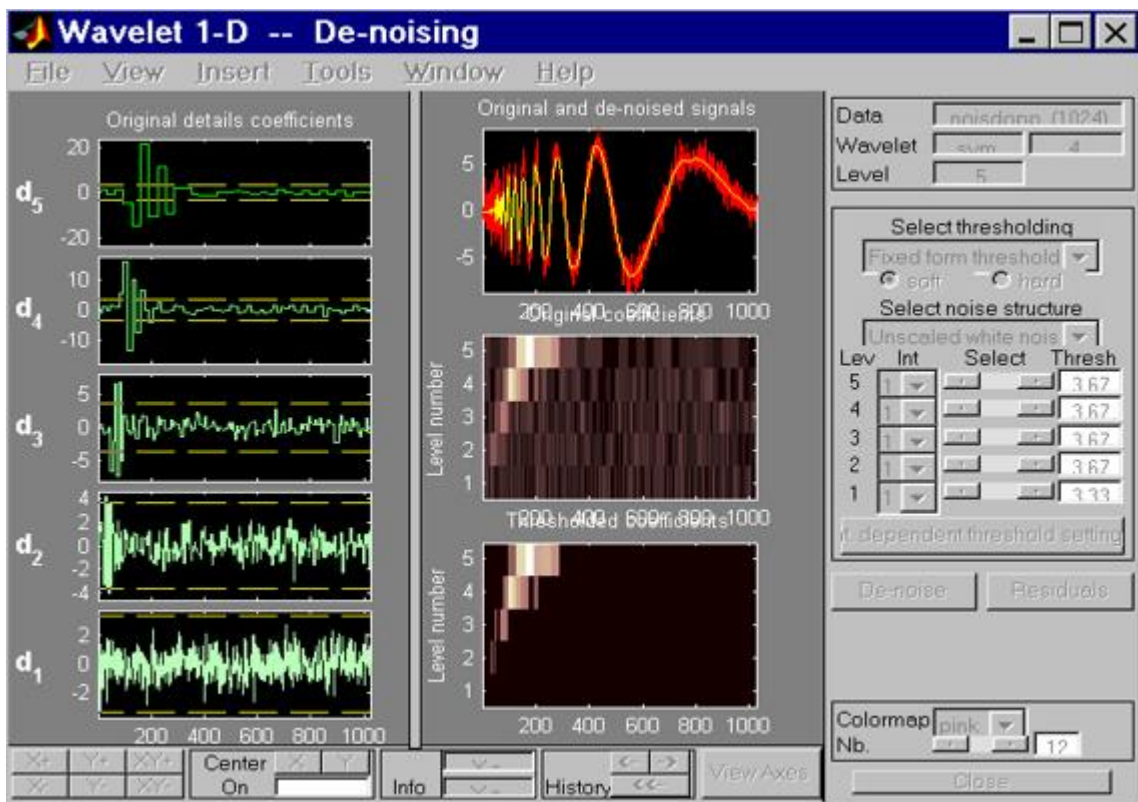


Рис. 3.8. Слайд 16 моделі вейвлет-перетворення векторних сигналів

Як видно, ця процедура зберігає самі великі коефіцієнти. Тут використовується м'який трешолдинг (рис. 3.8).

Слайд 19

Тут використовується жорсткий трешолдинг (рис. 3.9).

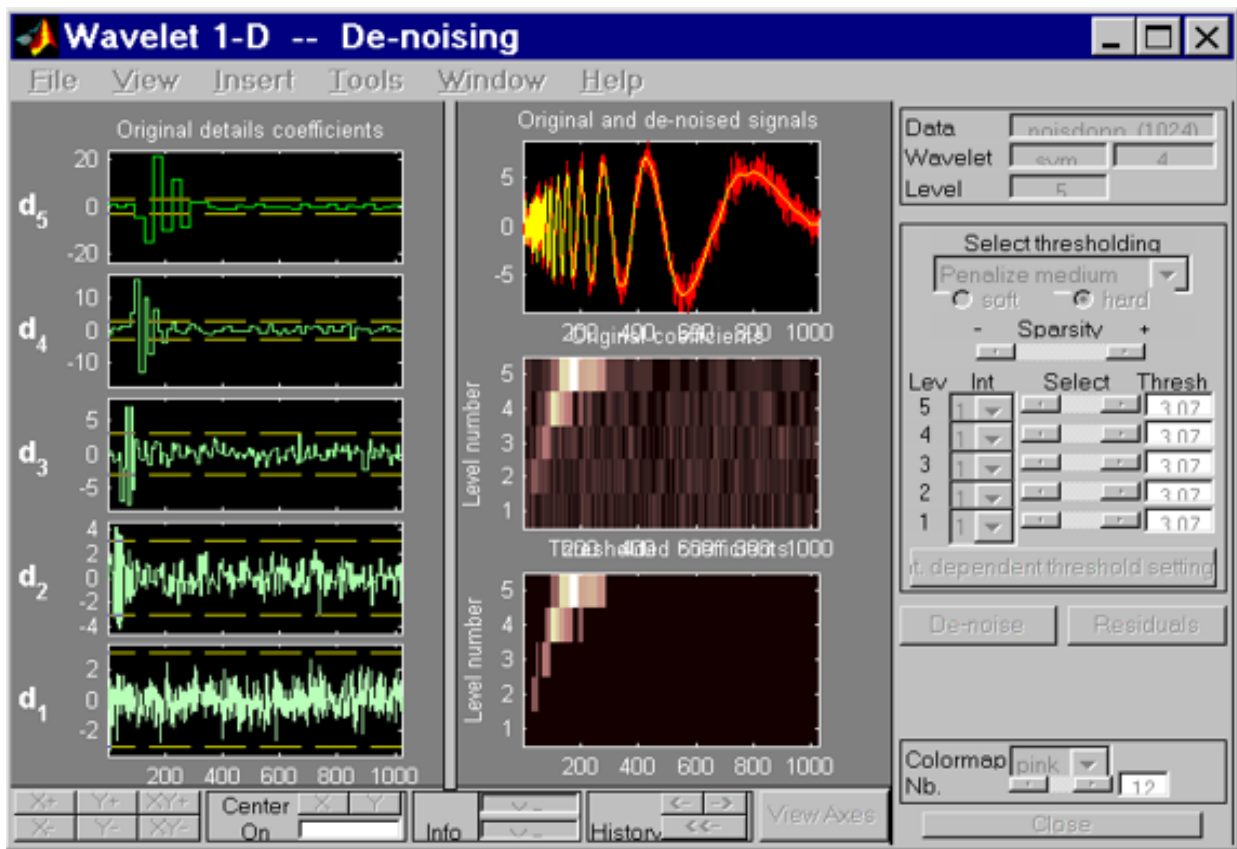


Рис. 3.9. Слайд 19 моделі вейвлет-перетворення векторних сигналів

Слайд 20

Нижче можна порівняти оригінальний сигнал з шумом і його відновлення з різними параметрами обробки.

Слайд 21

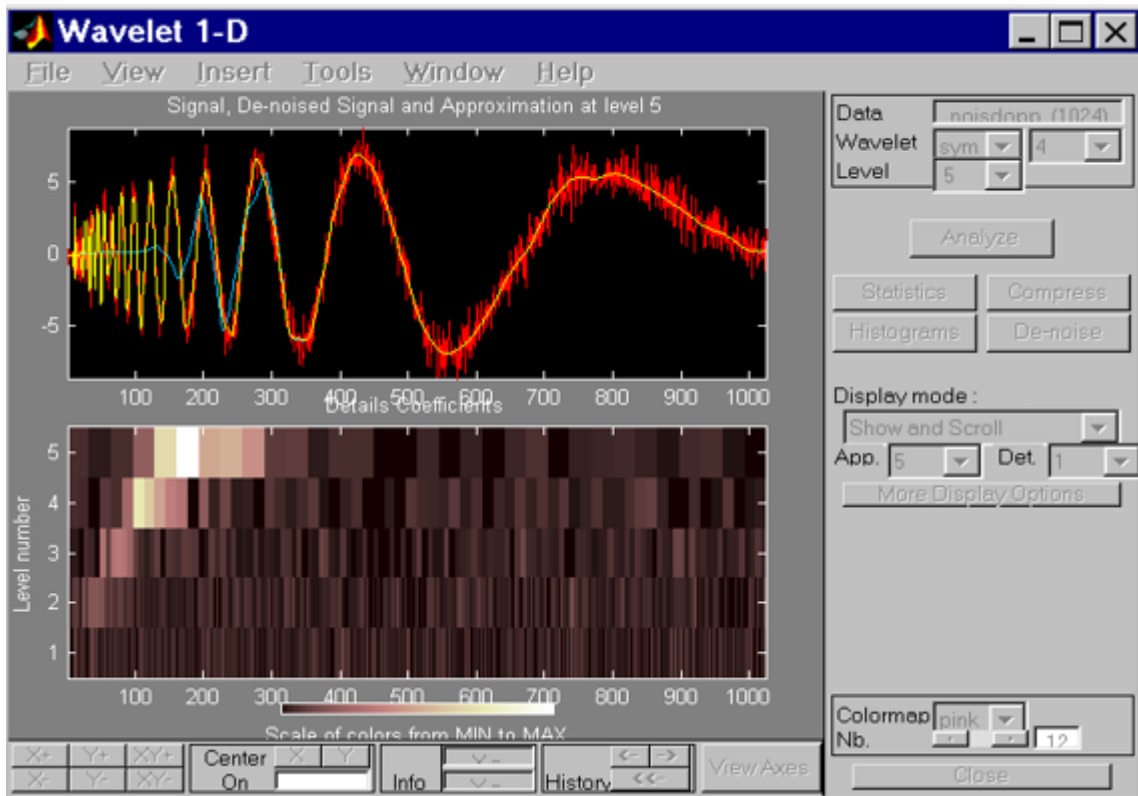


Рис. 3.10. Слайд 21 моделі вейвлет-перетворення векторних сигналів

3. Дослідження вейвлет-перетворення векторних сигналів.

3.1. Пройдіть шлях **Start/Toolboxes/Wavelet/Wavelet Toolbox Main**

Menu/Wavelet 1-D і перед Вами відкриється графічний інтерфейс ВВП векторних сигналів.

3.2. Виконайте стиск і фільтрацію шуму для наступних умов:

- трьох різних сигналів (меню **File/ Example Analysis: Basic Signals, Noisy Signals-Constant Noise Variance** і **Noisy Signals-Interval Dependent Noise Variance** – обирати сигнал **PN**-ий зверху, де **PN** – порядковий номер студента в списку групи);
- двох різних типів вейвлет;
- двох різних значень кількості рівнів розкладання;

- різних режимів стиску і фільтрації шуму.

3.3. Зафіксуйте не менш 6-ти пар відеокадрів для режимів, що розрізняються тільки однією умовою.

Вимоги до оформлення звіту

Звіт повинний містити:

1. Найменування і мету роботи.
2. Відеокадри моделювання з висновками по режимам, параметрам і якості, що досягається, обробки сигналів.

Контрольні питання

1. Що таке вейвлет-перетворення?
2. У чому полягає стиск сигналів без втрат?
3. Що таке трешолдинг сигналів?
4. Які переваги і недоліки вейвлет-перетворення?

ПРАКТИЧНА РОБОТА № 4

«БАНКИ ВЕЙВЛЕТ-ФІЛЬТРІВ»

Мета роботи: дослідження перетворень сигналів за допомогою банків вейвлет-фільтрів.

Програма роботи

1. Попередня підготовка до роботи.
2. Дослідження багаторівневого банку вейвлет-фільтрів.
3. Дослідження придушення шумів банком вейвлет-фільтрів.
4. Дослідження триканального вейвлет-трансмультіплектора.

Етапи роботи

1. Попередня підготовка до роботи

Використовуючи конспект лекцій і літературу засвоїти основи теорії вейвлет-перетворення сигналів і побудови банків фільтрів.

2. Дослідження багаторівневого банку вейвлет-фільтрів.

2.1. Пройдіть шлях **Start/Toolboxes/DSP/Demos/ Simulink Demos/Fixed Point/Wavelets Reconstruction**, виконайте **Open this model** і перед Вами відкриється імітаційна модель (рис. 4.1).

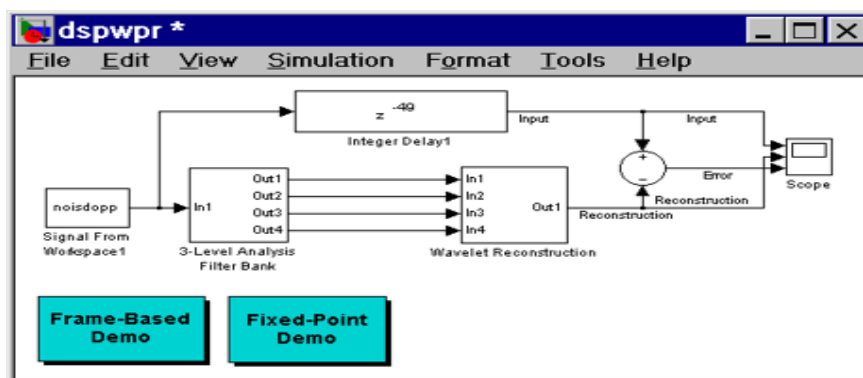


Рис. 4.1. Імітаційна модель банку вейвлет-фільтрів

Дана модель ілюструє здатність банку вейвлет-фільтрів до аналізу і синтезу сигналів (рис. 4.2).

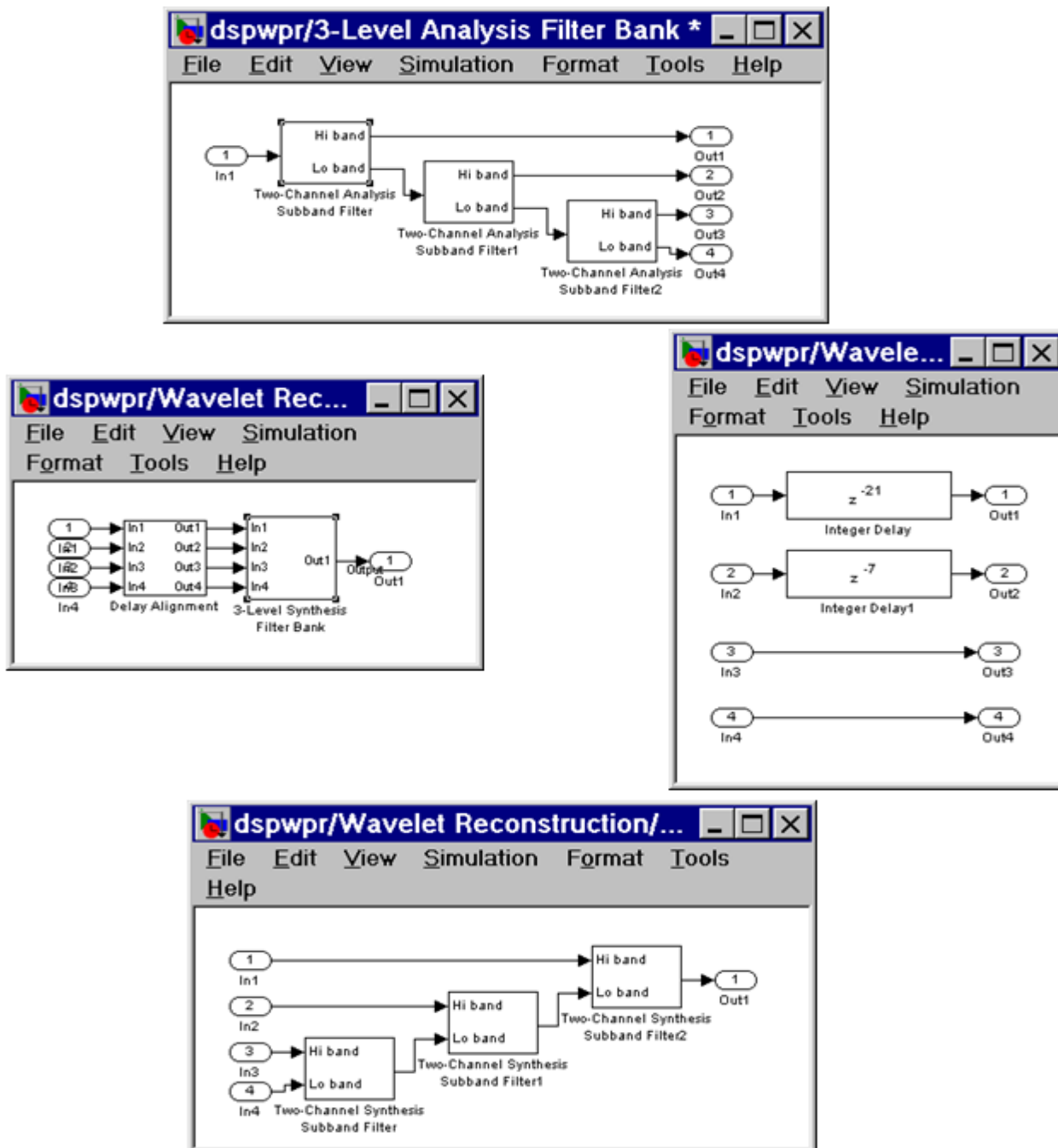


Рис. 4.2. Імітаційна модель банку вейвлет-фільтрів (детальний вигляд блоків)

2.2. Дослідіть шляхом моделювання структуру, параметри і результати роботи 3-рівневого банку вейвлет-фільтрів.

3. Дослідження придушення шумів банком вейвлет-фільтрів.

3.1. У шляху, зазначеному в п. 2.1, пройдіть до розділу **Denoising**, виконайте **Open this model** і перед Вами відкриється імітаційна модель (рис. 4.3).

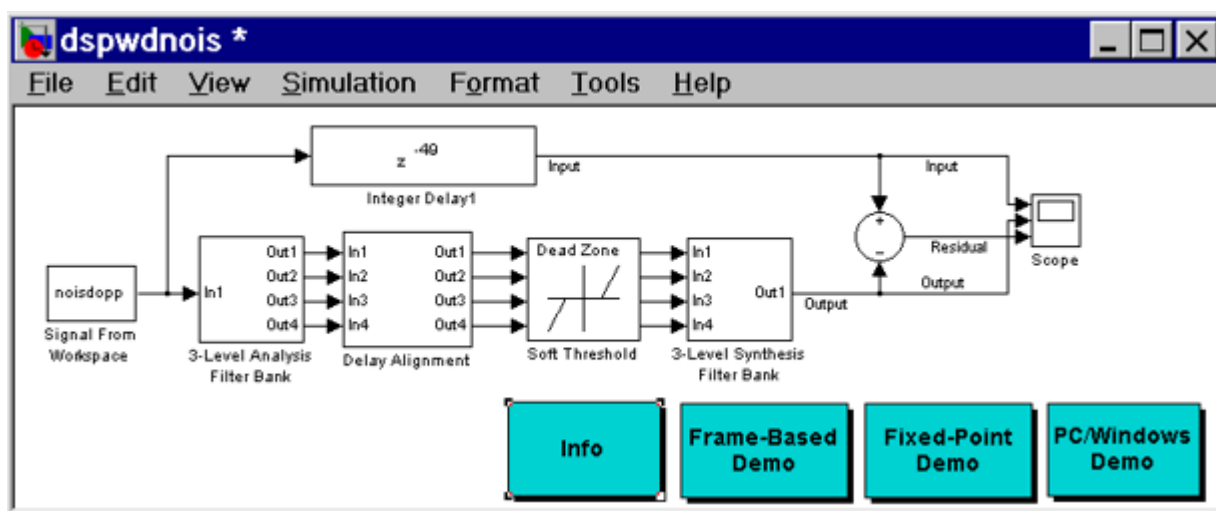


Рис. 4.3. Імітаційна модель банку вейвлет-фільтрів для фільтрації шумів

Дана модель дозволяє проводити фільтрацію шумів шляхом роздільного трешолдингу кожного з трьох рівнів розкладання сигналу і його грубої апроксимації. Вектор трешолдинга при повній реконструкції сигналу (без придушення шуму) дорівнює $\{0, 0, 0, 0\}$.

Імітаційна модель для PC/Windows дозволяє обробляти аудіосигнали в реальному масштабі часу (рис. 4.4).

3.2. Дослідіть вплив різних значень порогів (параметрів трешолдингу) на вхідний сигнал і доможіться найбільшої гладкості (найменшого шуму) функції на виході. Зафіксуйте відеокадри і відповідні параметри трешолдингу.

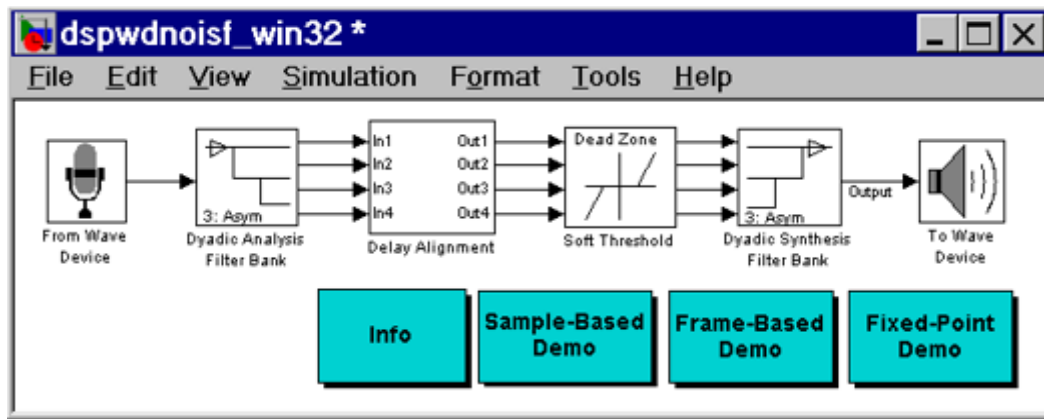


Рис. 4.4. Робота моделі в онлайн-режимі

4. Дослідження триканального вейвлет-трансмультіплексора.

4.1. У шляху, зазначеному в п. 2.1, пройдіть до розділу /Demos/Matlab Demos/Communications/Three-Channel Wavelet transmultiplexer (WTM), виконайте **Open this model** і перед Вами відкриється імітаційна модель (рис. 4.5)

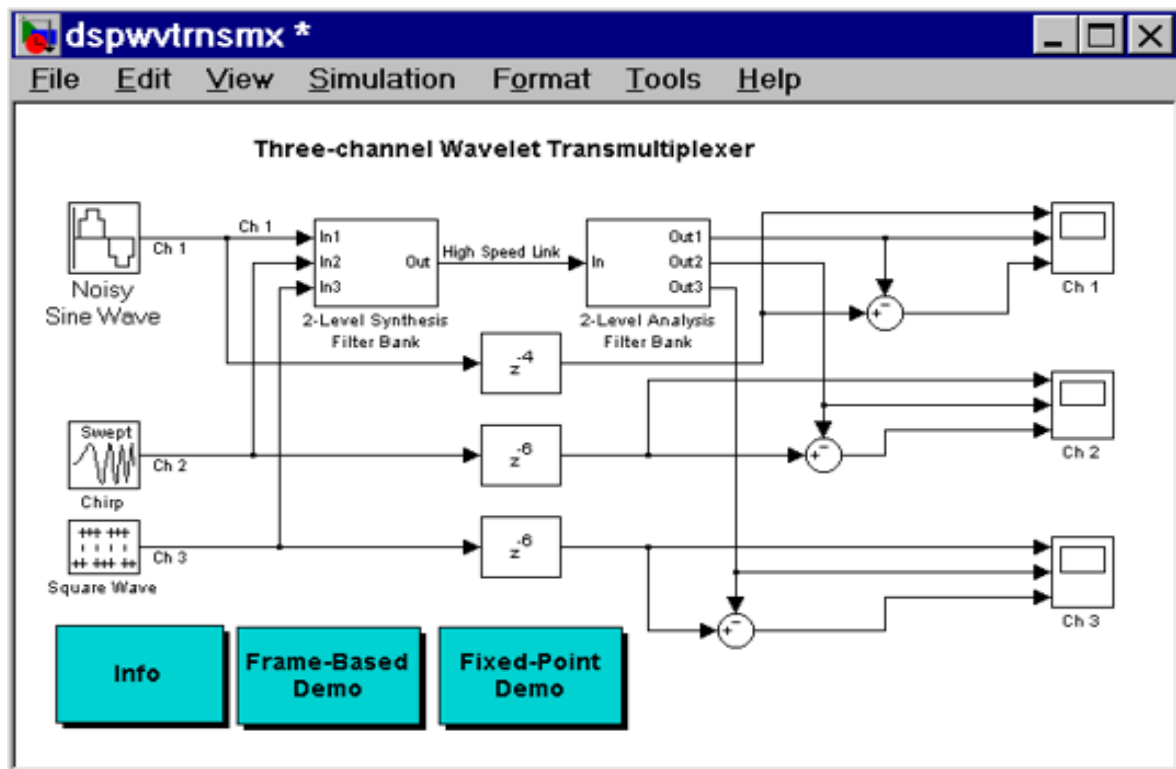


Рис. 4.5. Імітаційна модель вейвлет-трансмультіплексора

Дана імітаційна модель ілюструє ефективність вейвлет-трансмультіплектора (WTM) з дискретним вейвлет-перетворенням (ВВП) до відновлення трьох незалежних комбінованих сигналів, переданих по одній лінії зв'язку.

Цей WTM поєднує три вхідних сигнали для передачі по одній лінії зв'язку (зворотне ВВП), а потім розділяє (пряме ВВП) ці сигнали на прийомній стороні за допомогою банків вейвлет-фільтрів синтезу й аналізу.

Перевага WTM у порівнянні з частотним поділом каналів полягає, по-перше, у більш ефективному використанні смуги частот (немає необхідності виділяти для кожного сигналу окремий частотний канал із запасом по частоті) і, по-друге, у менших витратах на реалізацію.

4.2. Виконайте моделювання вейвлет-трансмультіплексування трьох сигналів:

- синусоїди з частотою $1/(50+PN*2)$ і рівнем шуму $0.1+PN/10$, де PN – порядковий номер студента в групі;
- сигналу розгорнення з кінцевою частотою $target=1/(PN*10)$;
- прямокутного імпульсу з періодом $100+PN*10$ і тривалістю імпульсу $20+PN*5$.

4.3. Проаналізуйте отримані результати і визначте: чому одночасні широкосмугові сигнали (з перекриттям по частоті і часу) розділяються в WTM?

Вимоги до оформлення звіту

Звіт повинний містити:

1. Найменування і мету роботи.
2. Відеокadри і параметри моделювання трешолдингу банком вейвлет-фільтрів.

3. Відеокадри вейвлет-трансмультіплексування трьох сигналів.
4. Висновки за отриманими результатами і відповідь на питання по п. 4.3.

Контрольні питання

1. Що таке банки фільтрів?
2. У чому полягає трансмультіплексування сигналів?
3. Що таке трешолдинг сигналів?
4. Які переваги і недоліки банків вейвлет-фільтрів?

КРИТЕРІЇ ОЦІНЮВАННЯ ПРАКТИЧНИХ РОБІТ

Подаються критерії оцінювання у вигляді переліку припущених недоліків, що знижують оцінку якості виконання практичного завдання.

Об'єктивна оцінка результатів розв'язання задач можлива (як і будь-яке інше вимірювання) лише при їх зіставленні з еталонними – зразками правильних та повних рішень (відповідей).

Оцінювання результатів практичних робіт здійснюється наступним чином:

1 питання – 2,5 бали;

2 питання – 2,5 бали;

3 питання – 2,5 бали;

4 питання – 2,5 бали.

Критеріями визначення оцінок приймається:

«Відмінно» – 9-10 балів;

«Добре» – 7,5-9 бали;

«Задовільно» – 6-7,5 бали;

«Незадовільно» – до 6 балів.

При остаточній оцінці результатів виконання завдання необхідно враховувати здатність студента:

- диференціювати, інтегрувати та уніфікувати знання;
- застосовувати правила, методи, принципи, закони у конкретних ситуаціях;
- інтерпретувати схеми, графіки, діаграми;
- аналізувати й оцінювати факти, події та спрогнозувати очікувані результати від прийнятих рішень;
- викладати матеріал на папері послідовно, з дотриманням вимог чинних стандартів.

РЕКОМЕНДОВАНІ ДЖЕРЕЛА ІНФОРМАЦІЇ

1. Корнієнко В.І. Інтелектуальне моделювання нелінійних динамічних процесів в керуванні, кібербезпеці, телекомунікаціях: підручник / В.І. Корнієнко, О.Ю. Гусєв, О.В. Герасіна. – Міністерство освіти і науки України, Національний технічний університет «Дніпровська політехніка». – Дніпро, НТУ «ДП», 2020. – 531 с.

2. Іванченко Г.Ф. Системи штучного інтелекту : навч. посібник / Г.Ф. Іванченко. – К., 2011. – 382 с.

3. Субботін С. О. Нейронні мережі : теорія та практика: навч. посібник / С.О. Субботін. – Житомир : Вид. О. О. Євенок, 2020. – 184 с.

4. Теорія систем управління: підручник / В.І. Корнієнко, О.Ю. Гусєв, О.В. Герасіна. – Міністерство освіти і науки України, Національний гірничий університет. – Дніпро, НГУ, 2017. – 497 с.

5. Гусєв О.Ю., Герасіна О.В., Корнієнко В.І., Нікольська О.І.. Основи теорії управління та адаптації: навч. посібн. - Дніпро: НТУ "ДП", 2021. - 345 с. - ISBN 978-966-350-755-2.

6. Gusev O.Yu. Theory of adaptive filtration: tutorial / O.Yu.Gusev, V.M.Gorev, V.I.Kornienko; Ministry of Education and Science of Ukrain, National Technical University “Dnipro polytechnic”.- Dnipro: NTU “DP”, 2019.- 156 p.

Навчальне видання

Корнієнко Валерій Іванович
Герасіна Олександра Володимирівна

СУЧАСНІ МЕТОДИ ОБРОБКИ ІНФОРМАЦІЇ

Методичні рекомендації до виконання практичних робіт
для здобувачів ступеня магістра
освітньо-професійної програми «Телекомунікації та радіотехніка»
зі спеціальності 172 Електронні комунікації та радіотехніка

У 2 частинах

Частина 2

Видано в авторській редакції

Електронний ресурс.
Підписано до видання 05.09.2024. Авт. арк. 3,0.

Національний технічний університет «Дніпровська політехніка»
49005, м. Дніпро, просп. Д. Яворницького, 19.