MINISTRY OF EDUCATION AND SCIENCE OF UKRAINE
NATIONAL TECHNICAL UNIVERSITY “DNIPRO POLYTECHNIC”

O.Yu. GUSEV

V.M. GOREV

V.I. KORNIIENKO

THEORY OF ADAPTIVE FILTRATION

Tutorial

Dnipro
NTU “DP”
2019

УДК 004.021
G 96

Рекомендовано вченою радою Національного технічного університету «Дніпровська політехніка» як навчальний посібник для студентів спеціальності 172 Телекомунікації та радіотехніка (протокол № 20 від 25.10.2019).

Рецензенти:
О.Й. Соколовський, д-р. фіз.-мат. наук, проф. (Дніпровський національний університет ім. Олеся Гончара);
М.О. Алексєєв, д.-р., техн. наук, проф. (Національний технічний університет «Дніпровська політехніка»).

Gusev O.Yu.
G 96 Theory of adaptive filtration: tutorial / O.Yu. Gusev, V.M. Gorev, V.I. Korniienko; Ministry of Education and Science of Ukraine, National Technical University "Dnipro polytechnic". – Dnipro : NTU "DP", 2019. – 156 p.

ISBN 978-966-350-715-6

Theoretical fundamentals of construction and parameters of adaptive filters as well as their practical use are considered. Attention is also paid to the structure and the use of frequency filters.

The tutorial is written in accordance with the education program for master students of the specialty of "Telecommunications and radiotechnics" and may be useful for scientists and specialists dealing with signal filtration theory.

Розглянуто теоретичні засади побудови і параметри адаптивних фільтрів, а також їхнє практичне застосування. Також приділено увагу структурі та використанню частотних фільтрів.

Текст видання відповідає програмі підготовки магістрів спеціальності «Телекомунікації та радіотехніка» і може бути корисним для науковців та спеціалістів, які займаються теорією фільтрації сигналів.

УДК 004.021

ISBN 978-966-350-715-6
© О.Ю. Гусєв, В.М. Горєв, В.І. Корнієнко, 2019
© НТУ «Дніпровська політехніка», 2019

Contents

Introduction

In the last thirty years significant contributions have been made in the signal processing field. The advances in digital circuit design have been the key technological development that sparked a growing interest in the field of digital signal processing. The resulting digital signal processing systems are attractive due to their low cost, reliability, accuracy, small physical sizes, and flexibility.

One example of a digital signal processing system is called filter. Filtering is a signal processing operation whose objective is to process a signal in order to manipulate the information contained in the signal. In other words, a filter is a device that maps its input signal to another output signal facilitating the extraction of the desired information contained in the input signal. A digital filter is the one that processes discrete-time signals represented in digital format. For time-invariant filters the internal parameters and the structure of the filter are fixed, and if the filter is linear, the output signal is a linear function of the input signal. Once prescribed specifications are given, the design of time-invariant linear filters entails three basic steps, namely: the approximation of the specifications by a rational transfer function, the choice of an appropriate structure defining the algorithm, and the choice of the form of implementation for the algorithm.

An adaptive filter is required when either the fixed specifications are unknown or the specifications cannot be satisfied by time-invariant filters. Strictly speaking, an adaptive filter is a nonlinear filter because its characteristics are dependent on the input signal and consequently the homogeneity and additivity conditions are not satisfied. However, if we freeze the filter parameters at a given instant of time, most adaptive filters considered in this text are linear in the sense that their output signals are linear functions of their input signals.

The adaptive filters are time-varying since their parameters are continually changing in order to meet a performance requirement. In this sense, we can interpret an adaptive filter as a filter that performs the approximation step on-line. Usually, the definition of the performance criterion requires the existence of a reference signal that is usually hidden in the approximation step of fixed-filter design. This discussion

brings the feeling that in the design of fixed (nonadaptive) filters a complete characterization of the input and reference signals is required in order to design the most appropriate filter that meets a prescribed performance. Unfortunately, this is not the usual situation encountered in practice, where the environment is not well defined. The signals that compose the environment are the input and the reference signals, and in cases where any of them is not well defined, the design procedure is to model the signals and subsequently design the filter. This procedure could be costly and difficult to implement on-line. The solution to this problem is to employ an adaptive filter that performs on-line updating of its parameters through a rather simple algorithm, using only the information available in the environment. In other words, the adaptive filter performs a data-driven approximation step.

The subject of this book is adaptive filtering, which is concerned with the choice of structures and algorithms for a filter that has its parameters (or coefficients) adapted in order to improve a prescribed performance criterion. The coefficient updating is performed using the information available at a given time.

It should be stressed that the derivation of the vector Kalman filter and Kalman–Bucy filter which are given in this book are not the same as the standard ones given in the literature intended for “pure” mathematicians. They are built on a simple analogy between the above-mentioned filters and the scalar Kalman filter, and they do not contain such difficult terms as a posteriori and a priori estimates.

Sections 1.1, 1.4, 1.5, 3.1–3.5, 4.1 and 4.2 are written by A. Yu. Gusev. Sections 1.2, 1.3, 3.6, 3.7, 3.9 and chapter 2 are written by V. N. Gorev. Sections 3.8, 3.10, 4.3 and 4.4 are written by V. I. Korniienko.

Chapter 1. General Information

1.1 Matrices. Determinants. Systems of linear equations

1.1.1 The concept of the matrix

A matrix of size $m \times n$, where m is the number of rows, n is the number of columns, is a table of numbers arranged in a certain order. The numbers themselves are called the **elements of the matrix**.

Matrices are denoted by capital Latin letters: *A, B, C*, ... Matrix elements are denoted by the symbol a_{ij}, where i is the row number, and j is the column number, at the intersection of which there is an element:

$$A = \begin{pmatrix} a_{11} & \cdots & a_{1n} \\ \vdots & \ddots & \vdots \\ a_{m1} & \cdots & a_{mn} \end{pmatrix}.$$

If the number of rows of the matrix is equal to the number of columns and is equal to n, then the matrix is called **square matrix of the order *n***.

If the number of rows of the matrix is not equal to the number of columns, then the matrix is called **rectangular**.

Depending on the values of the elements, the following matrices are distinguished:

1) **The zero** matrix is a matrix, all elements of which are 0:

$$\begin{pmatrix} 0 & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 \end{pmatrix}.$$

2) **Identity (unit) matrix** is the square matrix *E* or *I*, in which the elements of the main diagonal are equal to 1, and the remaining elements are equal to 0:

$$E = \begin{pmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 \end{pmatrix}.$$

3) **Diagonal** – diag (d_{11}, ..., d_{nn}), (its diagonal elements are d_{ij}, and non-diagonal are zero).

4) A square matrix (i.e., $i = j$) is called upper triangular if all its elements below the main diagonal are zero, respectively, a square matrix is called lower triangular if all its elements above the main diagonal are zero.

5) **The trace** of a square matrix is the sum of its diagonal elements

$$\mathrm{Tr}A = \sum_{i=1}^{n} a_{ii} .$$

6) The matrix B is called transposed to A and is denoted by $B = A^t$, if the rows of the matrix B are the columns of the matrix A with the same numbers (and the columns of B are the rows of A).

Example

$$\begin{pmatrix} 1 & -6 & 0 \\ 4 & 5 & -1 \end{pmatrix}^t = \begin{pmatrix} 1 & 4 \\ -6 & 5 \\ 0 & 1 \end{pmatrix}.$$

For any matrices A and B for which the product $A \cdot B$ is defined, the following equality exists:

$$\left(A \cdot B\right)^t = B^t A^t .$$

7) The matrix A is called **nonsingular** if its determinant is not equal to zero, i.e. $\det A \neq 0$.

8) **The inverse** of a **nonsingular** matrix A is a matrix A^{-1}, such that $A^{-1}A = E$.

9) A square matrix is called **symmetric** if $A = A^t$.

1.1.2 Matrix operations

Algebraic matrix operations include:

- addition (subtraction),

- multiplication by number,

- multiplication of the matrix by the matrix.

The sum (difference) of two matrices A and B of size $m \times n$ is matrix C of size $m \times n$, the elements of which are determined by the equations:

$$C = A \pm B, \;\; c_{ij} = a_{ij} \pm b_{ij}$$

where $i = 1, 2, \ldots, m$; $j = 1,2, \ldots, n$.

Thus, the addition (subtraction) operations of matrices are defined only if the matrices have the same size.

Example

$$\begin{pmatrix} 1 & 2 & 3 \\ 0 & 1 & -4 \end{pmatrix} + \begin{pmatrix} 2 & 1 & -5 \\ -2 & 3 & 0 \end{pmatrix} = \begin{pmatrix} 3 & 3 & -2 \\ -2 & 4 & -4 \end{pmatrix}.$$

The multiplication of the matrix A by the number λ is the matrix C, in which each element is equal to the product of the corresponding element of the matrix A by the number λ:

$$c_{ij} = \lambda a_{ij}.$$

Example

$$2 \cdot \begin{pmatrix} 1 & 2 & 3 \\ 0 & 1 & -4 \end{pmatrix} = \begin{pmatrix} 2 & 4 & 6 \\ 0 & 2 & -8 \end{pmatrix}.$$

The multiplication of matrix A of size $m \times n$ by matrix B of size $n \times k$ is the matrix C of size $m \times k$, in which the element standing at the intersection of the i-th row and j-th column is equal to the sum of the products of the elements of the i-th row of A by the elements of the j-th column of B factor, i.e.

$$c_{ij} = \sum_{s=1}^{n} a_{is} b_{sj} \ .$$

Thus, the multiplication operation of two matrices is defined if the number of columns of the first matrix is equal to the number of rows of the second.

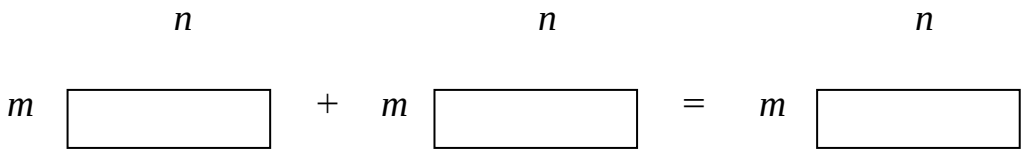
Remark. The matrix multiplication operation is not commutative:

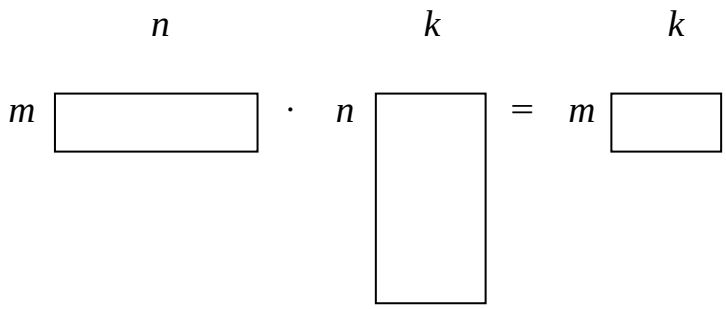
$$A \cdot B \neq B \cdot A.$$

Remark. If the products $A \cdot E$ and $E \cdot A$ are defined, then the following equalities take place:

$$A \cdot E = A, \quad E \cdot A = A.$$

An overview of matrix operations can be represented by the following scheme:





Exponentiation. The whole positive degree A^n $(n > 1)$ of a square matrix A is the product of n matrices equal to A, that is,

$$A^n = \underbrace{A \cdot A \cdot A \ldots \cdot A}_{n \text{ times}}.$$

By definition we have $A^0 = E$, $A^1 = A$. Obviously,

$$A^n A^k = A^{n+k}, \left(A^n\right)^k = A^{nk}.$$

1.1.3 The determinant

A determinant of a square matrix A is a number denoted by $|A|$, $\det(A)$ or Δ.

A rigorous definition of the determinant requires a detailed study of several concepts: substitutions and inversions, therefore we confine ourselves to a description of the methods for calculating the determinants of matrices of the 2nd and 3rd orders, and later on the determinants of any order.

The determinant of the 2nd order is the number calculated by the formula:

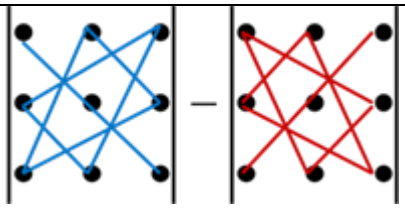
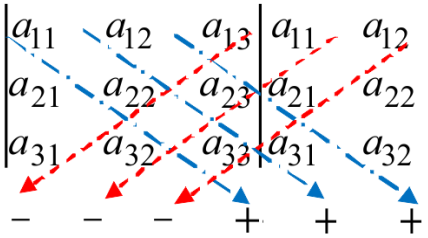
$$\begin{vmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{vmatrix} = a_{11}a_{22} - a_{12}a_{21},$$

that is, the 2nd order determinant is equal to the product of the elements on the main diagonal minus the product of the elements standing on the secondary diagonal.

The determinant of the 3rd order is the number calculated by the formula:

$$\begin{vmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{vmatrix} = a_{11}a_{22}a_{33} + a_{13}a_{21}a_{32} + a_{31}a_{12}a_{23} - a_{31}a_{22}a_{13} - a_{11}a_{32}a_{23} - a_{33}a_{21}a_{12}.$$

Of course, this formula is difficult to memorize, so you can use one of the rules to calculate the determinants:

Triangle rule	*Rule of Sarrus*
 Points which are the vertexes of the triangles denote the multipliers in the products	

Determinant properties

Property 1. The value of the determinant does not change when transposing:

$$\begin{vmatrix} a_{11} & \dots & a_{1n} \\ \vdots & \ddots & \vdots \\ a_{n1} & \cdots & a_{mn} \end{vmatrix} = \begin{vmatrix} a_{11} & \dots & a_{n1} \\ \vdots & \ddots & \vdots \\ a_{1n} & \cdots & a_{nn} \end{vmatrix}.$$

From property 1 it follows that the rows and columns of the determinant are equivalent, therefore all other properties will be formulated for the rows, but they will be valid for the columns as well.

Property 2. The determinant containing a row of all zeros is zero:

$$\begin{vmatrix} a_{11} & a_{12} & a_{13} & \dots & a_{1n} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ a_{n1} & a_{n2} & a_{n3} & \dots & a_{nn} \end{vmatrix} = 0.$$

The procedure in which two arbitrary rows (two arbitrary columns) of the determinant are swapped is called **a transposition of the rows**.

Property 3. The transposition of the rows changes the sign of the determinant to the opposite:

$$\begin{vmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ \vdots & \vdots & \vdots & \vdots \\ a_{i1} & a_{i2} & \cdots & a_{in} \\ \vdots & \vdots & \vdots & \vdots \\ a_{j1} & a_{j2} & \cdots & a_{jn} \\ \vdots & \vdots & \vdots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{vmatrix} = - \begin{vmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ \vdots & \vdots & \vdots & \vdots \\ a_{j1} & a_{j2} & \cdots & a_{jn} \\ \vdots & \vdots & \vdots & \vdots \\ a_{i1} & a_{i2} & \cdots & a_{in} \\ \vdots & \vdots & \vdots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{vmatrix}.$$

Property 4. If all the elements of any row of the determinant are multiplied by any number k, then the value of the determinant will change k times:

$$\begin{vmatrix} a_{11} & a_{12} & a_{13} & \dots & a_{1n} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ ka_{i1} & ka_{i2} & ka_{i3} & \dots & ka_{in} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ a_{n1} & a_{n2} & a_{n3} & \dots & a_{nn} \end{vmatrix} = k \begin{vmatrix} a_{11} & a_{12} & a_{13} & \dots & a_{1n} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ a_{i1} & a_{12} & a_{i3} & \dots & a_{in} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ a_{n1} & a_{n2} & a_{n3} & \dots & a_{nn} \end{vmatrix}.$$

Property 5. The value of the determinant will not change if we add another row multiplied by any number to some row:

$$\begin{vmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ \vdots & \vdots & \vdots & \vdots \\ a_{i1} & a_{i2} & \cdots & a_{in} \\ \vdots & \vdots & \vdots & \vdots \\ a_{j1} & a_{j2} & \cdots & a_{jn} \\ \vdots & \vdots & \vdots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{vmatrix} = \begin{vmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ \vdots & \vdots & \vdots & \vdots \\ a_{i1} & a_{i2} & \cdots & a_{in} \\ \vdots & \vdots & \vdots & \vdots \\ a_{j1} + ka_{i1} & a_{j2} + ka_{i2} & \cdots & a_{jn} + ka_{in} \\ \vdots & \vdots & \vdots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{vmatrix}.$$

Property 6. If each element of a certain row of determinant is represented as the sum of two terms, then the determinant can be represented as the sum of two determinants: in the first determinant the row with the same number contains first terms, and the second determinant – second terms:

$$\begin{vmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ \vdots & \vdots & \vdots & \vdots \\ b_{i1} + c_{i1} & b_{i2} + c_{i2} & \dots & b_{in} + c_{in} \\ \vdots & \vdots & \vdots & \vdots \\ a_{n1} & a_{n2} & \dots & a_{nn} \end{vmatrix} = \begin{vmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ \vdots & \vdots & \vdots & \vdots \\ b_{i1} & b_{i2} & \dots & b_{in} \\ \vdots & \vdots & \vdots & \vdots \\ a_{n1} & a_{n2} & \dots & a_{nn} \end{vmatrix} + \begin{vmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ \vdots & \vdots & \vdots & \vdots \\ c_{i1} & c_{i2} & \dots & c_{in} \\ \vdots & \vdots & \vdots & \vdots \\ a_{n1} & a_{n2} & \dots & a_{nn} \end{vmatrix}.$$

Property 7. The determinant having two equal rows is zero:

$$\begin{vmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ \vdots & \vdots & \vdots & \vdots \\ a_{i1} & a_{i2} & \cdots & a_{in} \\ \vdots & \vdots & \vdots & \vdots \\ a_{i1} & a_{i2} & \cdots & a_{in} \\ \vdots & \vdots & \vdots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{vmatrix} = 0.$$

Minors and algebraic adjuncts

The minor of a certain element of the determinant is the determinant obtained from the given one by crossing out the row and column at the intersection of which this element is located.

Example. Find the minors of elements a_{13} and a_{22} of the determinant

$$\begin{vmatrix} 3 & 8 & -2 \\ 1 & 4 & 0 \\ 7 & 5 & 1 \end{vmatrix}.$$

Solution. Minor of element $a_{13} = -2$: $\begin{vmatrix} 1 & 4 \\ 7 & 5 \end{vmatrix} = 5 - 28 = -23$;

Minor of element $a_{22} = 4$: $\begin{vmatrix} 3 & -2 \\ 7 & 1 \end{vmatrix} = 3 - (-14) = 17$.

The algebraic adjunct of some element of the determinant is the minor of this element multiplied by $(-1)^{i+j}$, where i is the row number, and j is the column number at the intersection of which the element is located. The algebraic adjunct of the element a_{ij} is denoted by A_{ij}.

Theorem. The determinant is equal to the sum of the products of elements of any row (or column) by their algebraic adjuncts.

$$\Delta = a_{i1}A_{i1} + a_{i2}A_{i2} + \ldots + a_{in}A_{in}.$$

Algorithm for finding the inverse matrix A^{-1}

1. Calculate the determinant $|A|$. If $|A| \neq 0$, then matrix A has an inverse. If $|A| = 0$, then matrix A has no inverse.

2. Find algebraic adjuncts to all elements of the matrix A.

3. Replace all elements of the matrix A with their algebraic adjuncts and transpose the resulting matrix (that is, swap rows and columns).

4. Divide all the elements of the resulting matrix by the determinant of the matrix A:

$$A^{-1} = \begin{pmatrix} \frac{A_{11}}{|A|} & \frac{A_{21}}{|A|} & \cdots & \frac{A_{n1}}{|A|} \\ \frac{A_{12}}{|A|} & \frac{A_{22}}{|A|} & \cdots & \frac{A_{n2}}{|A|} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{A_{1n}}{|A|} & \frac{A_{2n}}{|A|} & \cdots & \frac{A_{nn}}{|A|} \end{pmatrix}.$$

Comment. It can be shown that, for a non-singular second-order matrix $A = \begin{pmatrix} a & b \\ c & d \end{pmatrix}$, the inverse matrix is $A^{-1} = \frac{1}{|A|}\begin{pmatrix} d & -b \\ -c & a \end{pmatrix}$.

Elementary matrix transformations

Elementary transformations of the matrix are the following transformations:

1) transposition of rows (i.e., a procedure in which two arbitrary rows of the matrix are swapped);

2) multiplication of any row of the matrix by a non-zero number;

3) adding to any row of the matrix any other one multiplied by any number;

4) crossing out the row consisting of only zeros.

If the matrix B is obtained from A as a result of elementary transformations, then it is said that that A is **equivalent** to B: $A \sim B$.

A step-matrix is a matrix that satisfies the conditions:

1) if the *i*-th row is zero (that is, it consists of all zeros), then the (*i* + 1)-th row is also zero,

2) if the first nonzero elements of the *i*-th and (*i* + 1) -th rows are arranged in columns with the numbers *k* and *l*, respectively, then *k* < *l*.

Theorem. Any matrix can be brought to a stepwise form using elementary transformations.

Proof. Let us consider a matrix $\begin{pmatrix} a_{11} & a_{12} & a_{13} & \dots & a_{1n} \\ a_{21} & a_{22} & a_{23} & \dots & a_{2n} \\ a_{31} & a_{32} & a_{33} & \dots & a_{3n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ a_{k1} & a_{k2} & a_{k3} & \dots & a_{kn} \end{pmatrix}$. To the *i*-th row of the matrix (*i* = 2, 3, 4, ..., *k*) we add the first row multiplied by $-\frac{a_{i1}}{a_{11}}$, so we obtain the matrix

$$\begin{pmatrix} a_{11} & a_{12} & a_{13} & \dots & a_{1n} \\ 0 & a'_{22} & a'_{23} & \dots & a'_{2n} \\ 0 & a'_{32} & a'_{33} & \dots & a'_{3n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & a'_{k2} & a'_{k3} & \dots & a'_{kn} \end{pmatrix}, \quad a'_{ij} = a_{ij} - \frac{a_{i1}}{a_{11}} a_{1j}.$$

In this matrix, to the *i*-th row (*i* = 3, 4, ..., k) we add the second row multiplied by $-\frac{a'_{i2}}{a'_{22}}$, thereby obtaining the matrix:

$$\begin{pmatrix} a_{11} & a_{12} & a_{13} & \dots & a_{1n} \\ 0 & a'_{22} & a'_{23} & \dots & a'_{2n} \\ 0 & 0 & a''_{33} & \dots & a''_{3n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & a''_{k3} & \dots & a''_{kn} \end{pmatrix}, \quad a''_{ij} = a'_{ij} - \frac{a'_{i2}}{a'_{22}} a'_{2j}.$$

Then to the *i*-th row (*i* = 4, ..., n) we add the third row multiplied by $-\frac{a''_{i3}}{a''_{33}}$, continuing this process, we arrive at a step matrix.

1.1.4 Systems of linear equations

1.1.4.1 General information about systems of linear equations

Consider a system of *n* linear equations with *n* unknowns:

$$\begin{cases} a_{11}x_1 + a_{12}x_2 + ... + a_{1n}x_n = b_1 \\ a_{21}x_1 + a_{22}x_2 + ... + a_{2n}x_n = b_2 \\ \vdots \\ a_{n1}x_1 + a_{n2}x_2 + ... + a_{nn}x_n = b_n \end{cases}. \qquad (1.1)$$

The numbers a_{ij} (*i, j* = 1, 2, ..., *n*) are called **the coefficients** of the system, and the numbers b_j are called the **free terms**.

The solution of the system is the set of numbers $x_1, x_2, ..., x_n$, by substitution of which into the system we get the correct equalities.

A system that has a solution is called **a joint**. A system that does not have a solution is called **disjoint**. If all free terms are equal to 0, then the system is called **homogeneous**, otherwise - **inhomogeneous**.

Comment. A homogeneous system of linear equations is always joint. It necessarily has a zero solution (maybe not the only one).

1.1.4.2 Matrix notation of a system of linear equations

Let's introduce the following notations:

$$A = \begin{pmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{pmatrix}, \; X = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix}, \; B = \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{pmatrix}.$$

The system (1.1) can be written in the following matrix form:

$$A \cdot X = B.$$

If $\det(A) \neq 0$, then A^{-1} exists, and by multiplying both sides of the equation by A^{-1} we obtain

$$\underbrace{A^{-1} \cdot A}_{=E} \cdot X = A^{-1} \cdot B \Rightarrow E \cdot X = A^{-1} \cdot B \Rightarrow X = A^{-1} \cdot B.$$

1.1.4.3 Cramer Formulas

It was shown above that the solution of system (1.1) if $\det(A) \neq 0$ can be found in the form $X = A^{-1}B$, which explicitly gives

$$X = A^{-1}B = \begin{pmatrix} \frac{1}{|A|}(A_{11}b_1 + A_{21}b_2 + \ldots + A_{n1}b_n) \\ \frac{1}{|A|}(A_{12}b_1 + A_{22}b_2 + \ldots + A_{n2}b_n) \\ \vdots \\ \frac{1}{|A|}(A_{1n}b_1 + A_{2n}b_2 + \ldots + A_{nn}b_n) \end{pmatrix},$$

so $x_i = \frac{1}{|A|}(A_{1i}b_1 + A_{2i}b_2 + \ldots + A_{ni}b_n) = \frac{\Delta_i}{\Delta}$ where $\Delta = |A|$ is the determinant of the matrix A, and $\Delta_i = \begin{vmatrix} a_{11} & \cdots & a_{1,i-1} & b_1 & a_{1,i+1} & \cdots & a_{1n} \\ a_{21} & \cdots & a_{2,i-1} & b_2 & a_{2,i+1} & \cdots & a_{2n} \\ \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ a_{n1} & \cdots & a_{n,i-1} & b_n & a_{n,i+1} & \cdots & a_{nn} \end{vmatrix}$ is the determinant, obtained from the determinant Δ by replacing the i-th column with a column of free terms.

Thus, the obtained results

$$x_i = \frac{\Delta_i}{\Delta};\ i = 1, 2, \ldots n.$$

are called **Cramer formulas**.

1.1.4.4 Gauss method (method of elimination of unknowns)

Let us consider a system of *k* linear equations with *n* unknowns:

$$\begin{cases} a_{11}x_1 + a_{12}x_2 + \ldots + a_{1n}x_n = b_1 \\ a_{21}x_1 + a_{22}x_2 + \ldots + a_{2n}x_n = b_2 \\ \vdots \\ a_{k1}x_1 + a_{k2}x_2 + \ldots + a_{kn}x_n = b_k \end{cases}. \qquad (1.2)$$

Let's write the matrix of this system:

$$A = \begin{pmatrix} a_{11} & a_{12} & \ldots & a_{1n} \\ a_{21} & a_{22} & \ldots & a_{2n} \\ \ldots & \ldots & \ldots & \ldots \\ a_{k1} & a_{k2} & \ldots & a_{kn} \end{pmatrix}.$$

The matrix, obtained from matrix *A* as a result of attributing the column of free terms to the right is called **the extended** matrix of the system and is denoted by $\overline{A}$:

$$\overline{A} = \begin{pmatrix} a_{11} & a_{12} & \ldots & a_{1n} & b_1 \\ a_{21} & a_{22} & \ldots & a_{2n} & b_2 \\ \ldots & \ldots & \ldots & \ldots & \ldots \\ a_{k1} & a_{k2} & \ldots & a_{kn} & b_k \end{pmatrix}.$$

The Gauss method is as follows. First of all we should eliminate the variable x_1 from all equations of the system, starting with the second. Then we should eliminate the variable x_2 from all equations of the system, starting with the third, and so on. In other words, we bring the expanded matrix of the system to a stepwise form.

If the number of equations of the system is equal to the number of unknowns and is equal to *n*, then finally we obtain a system

$$\begin{cases} a_{11}x_1 + a_{12}x_2 + \ldots + a_{1n}x_n = b_1 \\ a'_{22}x_2 + \ldots + a'_{2n}x_n = b'_2 \\ \vdots \\ a'''_{nn}x_n = b'''_n \end{cases},$$

from which it is easy to find the variables $x_n, x_{n-1},..., x_2, x_1$ consistently.

1.2. Random processes and correlation functions

A random quantity is a quantity that can take different values with different probabilities. A function $x(t)$ of a real variable t is a **random function** if at each value of the variable t $x(t)$ is a random quantity. If the variable t is time, the random function $x(t)$ is a **random process**.

The one-dimensional probability density of a random process $x(t)$ is the quantity $\omega_1(x_1,t_1)$ such that $\omega(x_1,t_1)dx_1$ is the probability of $x(t_1)$ to take values between x_1 and $x_1 + dx_1$, i.e.

$$\omega_1(x_1,t_1)dx_1 = P\left(x(t_1) \in (x_1, x_1 + dx_1)\right). \qquad (1.3)$$

The two-dimensional probability density of a random process $x(t)$ is the quantity $\omega_2(x_1,t_1;x_2,t_2)$ such that $\omega(x_1,t_1;x_2,t_2)dx_1dx_2$ is the probability of the fact that $x(t_1) \in (x_1, x_1 + dx_1)$ and at the same time $x(t_2) \in (x_2, x_2 + dx_2)$:

$$\omega_2(x_1,t_1;x_2,t_2)dx_1dx_2 = P\left(x(t_1) \in (x_1, x_1 + dx_1), x(t_2) \in (x_2, x_2 + dx_2)\right). \qquad (1.4)$$

From (1.4) it is obvious that

$$\omega_2(x_1,t_1;x_2,t_2) = \omega_2(x_2,t_2;x_1,t_1). \qquad (1.5)$$

Similarly, the **n-dimensional probability density** of a random process $x(t)$ is the quantity $\omega_n(x_1,t_1;x_2,t_2;...;x_n,t_n)$ such that

$$\omega_n(x_1,t_1;x_2,t_2;...;x_n,t_n)dx_1dx_2...dx_n = \\ = P\left(x(t_1) \in (x_1, x_1 + dx_1), x(t_2) \in (x_2, x_2 + dx_2),..., x(t_n) \in (x_n, x_n + dx_n)\right). \qquad (1.6)$$

The probability density satisfies the normalization:

$$\int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty}\ldots\int_{-\infty}^{+\infty} dx_1 dx_2 \ldots dx_n \omega_n\left(x_1,t_1;x_2,t_2;\ldots;x_n,t_n\right)=1, \tag{1.7}$$

whose physical meaning is that the sum of all possible probabilities is equal to 1.

Generally speaking, a random process is described by the probability densities of all dimensions. However, it is very hard to work with probability densities of high dimensions. That is why random processes are often described only by the one- and the two-dimensional probability density. Moreover, there are a lot of random processes that are rigorously described only by the one- and the two-dimensional probability density, and the probability densities of all the other dimensions can be described in terms of the one and the two-dimensional one.

The mathematical expectation (the average value) of a random process $x(t)$ is the quantity $\langle x(t)\rangle$:

$$\langle x(t)\rangle = m_x(t) = \int_{-\infty}^{+\infty} dx x \omega_1(x,t). \tag{1.8}$$

The average value of a function of a random process is defined as

$$\langle f(x(t))\rangle = \int_{-\infty}^{+\infty} dx f(x)\omega_1(x,t). \tag{1.9}$$

For example, the mean square of a random process according to (1.9) is

$$\langle x^2(t)\rangle = \int_{-\infty}^{+\infty} dx x^2 \omega_1(x,t). \tag{1.10}$$

It should be noticed that if the values of a random process $x(t)$ are taken at different instants, then the average value of the corresponding product is defined in terms of the two-dimensional probability density rather than in terms of the one-dimensional one:

$$\langle x(t_1)x(t_2)\rangle = \int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} dx_1 dx_2 x_1 x_2 \omega_2(x_1,t_1;x_2,t_2). \tag{1.11}$$

The variance $D_x(t)$ and **the mean-square deviation** $\sigma_x(t)$ of a random process $x(t)$ are defined as

$$D_x(t)=\left\langle x^2(t)\right\rangle-\left\langle x(t)\right\rangle^2,\ \sigma(t)=\sqrt{D_x(t)}. \tag{1.12}$$

The correlation function is introduced in the literature by different ways. In some books, the correlation function $R_x(t_1,t_2)$ of a random process $x(t)$ is defined as $R_x(t_1,t_2)=\left\langle x(t_1)x(t_2)\right\rangle$, in others it is defined as $R_x(t_1,t_2)=\left\langle x(t_1)x(t_2)\right\rangle-$ $-\left\langle x(t_1)\right\rangle\left\langle x(t_2)\right\rangle$. It should be noticed that if the average value $\left\langle x(t)\right\rangle=0$ for each t, then these definitions coincide with each other, but they do not coincide with each other in the general case. For definiteness, we define the **correlation function** of a random process $x(t)$ as

$$R_x(t_1,t_2)=\left\langle x(t_1)x(t_2)\right\rangle. \tag{1.13}$$

A random process is **stationary** if for the probability densities *of all the dimensions* the following property is valid:

$$\forall\tau\ \ \omega_n(x_1,t_1;x_2,t_2;...;x_n,t_n)=\omega_n(x_1,t_1+\tau;x_2,t_2+\tau;...;x_n,t_n+\tau), \tag{1.14}$$

i.e. if we add the same value to all the time arguments of the probability density, the probability density does not change. Such processes are often called **strictly stationary** processes.

However, as mentioned before, for simplicity random processes are often described in terms of only the one- and the two-dimensional probability density. A random process is a **wide sense stationary** random process if the property (1.14) is valid only for $n=1,2$. Wide sense stationary processes are also called **weakly stationary processes**. If a random process is strictly stationary, it is also wide sense stationary, but, generally speaking, the converse is not true.

Let us show that **the average value of a stationary random process is time independent**. Let us consider the average values at two arbitrarily taken different instants $m_x(t_1)$ and $m_x(t_2)$. By the definition of (1.8)

$$m_x(t_1)=\int_{-\infty}^{+\infty} dx x\omega_1(x,t_1),\ m_x(t_2)=\int_{-\infty}^{+\infty} dx x\omega_1(x,t_2). \tag{1.15}$$

Choosing $\tau = t_2 - t_1$ in (1.12), we obtain

$$\omega_1(x_1,t_1)=\omega_1(x_1,t_1+(t_2-t_1))=\omega_1(x_1,t_2). \tag{1.16}$$

From (1.15) and (1.16) we have

$$m_x(t_1)=m_x(t_2). \tag{1.17}$$

So the average value of a stationary process is the same for two arbitrarily taken different instants, that is why it is time independent.

Now let us show that **the correlation function** $R_x(t_1,t_2)$ **of a stationary process** $x(t)$ **depends only on the time difference** $t_2 - t_1$. From (1.11) and (1.13) we have

$$R_x(t_1,t_2)=\int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} dx_1 dx_2 x_1 x_2 \omega_2(x_1,t_1;x_2,t_2). \tag{1.18}$$

Choosing $\tau = -t_2$ in (1.14), we obtain

$$\omega_2(x_1,t_1;x_2,t_2)=\omega_2(x_1,t_1-t_2;x_2,0). \tag{1.19}$$

From (1.19) and (1.18) we have

$$R_x(t_1,t_2)=\int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} dx_1 dx_2 x_1 x_2 \omega_2(x_1,t_1-t_2;x_2,0)=R_x(t_1-t_2), \tag{1.20}$$

which was to be proved. From (1.20) it can be seen that **the correlation function of a stationary random process is a one-variable function rather than a two-variable one;** from (1.20) it can be seen that

$$R_x(\tau)=\int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} dx_1 dx_2 x_1 x_2 \omega_2(x_1,\tau;x_2,0). \tag{1.21}$$

Choosing $\tau = t$ in (1.14), we obtain

$$\omega_2(x_1,\tau;x_2,0)=\omega_2(x_1,\tau+t;x_2,t), \tag{1.22}$$

and from (1.22) and (1.21) we have

$$R_x(\tau)=\int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty}dx_1dx_2x_1x_2\omega_2(x_1,\tau+t;x_2,t)=\langle x(t+\tau)x(t)\rangle. \tag{1.23}$$

It should be noticed that **the result (1.23) is widely used**.

Let us show that **the correlation function of a stationary process is an even one**. From (1.21) we have

$$R_x(-\tau)=\int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty}dx_1dx_2x_1x_2\omega_2(x_1,-\tau;x_2,0). \tag{1.24}$$

From (1.14) we have

$$\omega_2(x_1,-\tau;x_2,0)=\omega_2(x_1,-\tau+\tau;x_2,0+\tau)=\omega_2(x_1,0;x_2,\tau), \tag{1.25}$$

which with account for (1.24) gives

$$\begin{aligned}R_x(-\tau)&=\int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty}dx_1dx_2x_1x_2\omega_2(x_1,0;x_2,\tau)=\{x_1\leftrightarrow x_2\}=\\&=\int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty}dx_2dx_1x_2x_1\omega_2(x_2,0;x_1,\tau)=\int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty}dx_1dx_2x_1x_2\omega_2(x_2,0;x_1,\tau).\end{aligned} \tag{1.26}$$

From (1.26) and (1.5) we obtain

$$R_x(-\tau)=\int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty}dx_1dx_2x_1x_2\omega_2(x_1,\tau;x_2,0)=R_x(\tau), \tag{1.27}$$

which was to be proved.

The cross-correlation function of two random processes $x(t)$ and $y(t)$ is the function $R_{xy}(t_1,t_2)$ defined as

$$R_{xy}(t_1,t_2)=\langle x(t_1)y(t_2)\rangle=\int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty}dxdyxy\omega(x,t_1;y,t_2) \tag{1.28}$$

where $\omega(x_1,t_1;y_2,t_2)dx_1dy_2$ is the probability of the fact that $x(t_1)\in(x_1,x_1+dx_1)$ and at the same time $y(t_2)\in(y_2,y_2+dy_2)$. Obviously,

$$\omega(x,t_1;y,t_2)=\omega(y,t_2;x,t_1). \tag{1.29}$$

It should be stressed that in the literature the cross-correlation function is not always defined as in (1.28). The definition $R_{xy}(t_1,t_2)=\langle x(t_1)y(t_2)\rangle-\langle x(t_1)\rangle\langle y(t_2)\rangle$ is also often used, but for definiteness, we will use the definition (1.28). These two definitions coincide in the case where $x(t)$ or $y(t)$ is a stationary random process with a zero average value.

If both processes $x(t)$ and $y(t)$ are stationary, then it is often assumed that the probability density $\omega(x,t_1;y,t_2)$ obeys the property

$$\forall\tau\ \ \omega(x,t_1+\tau;y,t_2+\tau)=\omega(x,t_1;y,t_2). \tag{1.30}$$

In such a case, **the cross-correlation function of two stationary processes obeys the following properties**:

$$R_{xy}(t_1,t_2)=R_{xy}(t_1-t_2),\ R_{xy}(\tau)=\langle x(t+\tau)y(t)\rangle,\ R_{xy}(\tau)=R_{yx}(-\tau). \tag{1.31}$$

Let us prove them. From (1.30) we have

$$\omega(x,t_1;y,t_2)=\omega(x,t_1-t_2;y,0). \tag{1.32}$$

From (1.32) and (1.28) it can be seen that

$$R_{xy}(t_1,t_2)=\int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty}dxdyxy\omega(x,t_1-t_2;y,0)=R_{xy}(t_1-t_2), \tag{1.33}$$

so the first property in (1.31) is proved. As can be seen from (1.33), the cross-correlation function of two stationary processes is a one-variable function rather than a two-variable one:

$$R_{xy}(\tau)=\int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty}dxdyxy\omega(x,\tau;y,0). \tag{1.34}$$

From (1.30) we obtain

$$\omega(x,\tau;y,0)=\omega(x,t+\tau;y,t), \tag{1.35}$$

which with account for (1.34) leads to

$$R_{xy}(\tau)=\int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty}dxdyxy\omega(x,t+\tau;y,t)=\langle x(t+\tau)y(t)\rangle, \tag{1.36}$$

so the second property in (1.31) is proved. From (1.34) we see that

$$R_{yx}(-\tau)=\int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty}dxdyxy\omega(y,-\tau;x,0). \tag{1.37}$$

From (1.30) and (1.29) we have

$$\omega(y,-\tau;x,0)=\omega(y,-\tau+\tau;x,\tau)=\omega(y,0;x,\tau)=\omega(x,\tau;y,0), \tag{1.38}$$

so with account for (1.38), (1.37) and (1.34) we have

$$R_{xy}(\tau)=R_{yx}(-\tau). \tag{1.39}$$

So the third property in (1.31) is proved.

It should be stressed that the above-mentioned definitions of average values are **assembly average** definitions. The **time average** definition for a realization of a stationary process $x(t)$ is also used:

$$\langle x(t)\rangle_t=\lim_{T\to\infty}\frac{1}{2T}\int_{-T}^{+T}dtx(t). \tag{1.40}$$

In fact, the definition (1.40) is similar to the well-known definition of the average value of a function over the whole time axis. The definition of (1.40) is valid if the corresponding limits exist.

For stationary processes, a time correlation function, $\Re_x(\tau)$, and a time cross-correlation function, $\Re_{xy}(\tau)$, are introduced, the idea of their definitions is based on (1.23) and (1.36):

$$\Re_x(\tau)=\lim_{T\to\infty}\frac{1}{2T}\int_{-T}^{+T}dtx(t)x(t+\tau),\ \Re_{xy}(\tau)=\lim_{T\to\infty}\frac{1}{2T}\int_{-T}^{+T}dtx(t+\tau)y(t), \tag{1.41}$$

the definitions (1.41) are valid if the corresponding limits exist.

It should be stressed that the time average may be different for different realizations of a random process. Moreover, the time average may not coincide with

the assembly average. But there exists a wide class of stationary random processes called **erdogic** ones. The time average and the assembly average coincide in the case of an ergodic process. For ergodic processes the time correlation function coincides with the correlation function and the time cross-correlation function coincides with the cross-correlation function, i.e. **for ergodic processes**

$$\left\langle x(t)\right\rangle_t = \left\langle x(t)\right\rangle \ , \ \Re_x(\tau) = R_x(\tau), \ \Re_{xy}(\tau) = R_{xy}(\tau). \tag{1.42}$$

1.3. Spectral density of a stationary ergodic random process

Consider a stationary egrodic random process $x(t)$ whose correlation function is $R_x(t)$. Then by definition the **spectral density** of this process is the Fourier transform $S_x(\omega)$ of the correlation function $R_x(t)$:

$$S_x(\omega) = \int_{-\infty}^{+\infty} dt R_x(t) e^{-i\omega t} \ . \tag{1.43}$$

The correlation function can be expressed in terms of the spectral density by the inverse Fourier transform:

$$R_x(t) = \frac{1}{2\pi} \int_{-\infty}^{+\infty} dt S_x(\omega) e^{i\omega t} \ . \tag{1.44}$$

Let us show that $S_x(\omega)$ is an even function of ω. Let us rewrite (1.43) as

$$S_x(\omega) = \int_{-\infty}^{+\infty} dt R_x(t)\cos(\omega t) - i\int_{-\infty}^{+\infty} dt R_x(t)\sin(\omega t). \tag{1.45}$$

The second integral on the right-hand side of (1.45) vanishes because $R_x(t)$ is an even function of time, $\sin(\omega t)$ is an odd function of time and the corresponding integrand is an odd function of time. So, only the first integral on the right-hand side of (1.45) "survives". Then it is obvious that $S_x(\omega)$ is an even function of ω because $\cos(\omega t)$ is an even function of ω.

Now let us illustrate the physical meaning of the spectral density. Let we have a resistor r across which a voltage $U(t)$ is applied, $U(t)$ is a stationary random process. Then the active power on the resistor is

$$P(t)=U^2(t)/r, \tag{1.46}$$

the average value of this power is

$$\langle P(t)\rangle=\frac{1}{r}\langle U^2(t)\rangle, \tag{1.47}$$

and the correlation function of $U(t)$ is

$$R_U(\tau)=\langle U(t+\tau)U(t)\rangle \Rightarrow R_U(0)=\langle U^2(t)\rangle. \tag{1.48}$$

So (1.47) can be rewritten as

$$\langle P(t)\rangle=\frac{1}{r}R_U(0). \tag{1.49}$$

On the basis of (1.44) we have

$$R_U(0)=\frac{1}{2\pi}\int_{-\infty}^{+\infty}S_U(\omega)d\omega=\frac{1}{\pi}\int_{0}^{+\infty}S_U(\omega)d\omega, \tag{1.50}$$

the last equality in (1.50) is due to the fact that $S_U(\omega)$ is an even function of ω. So from (1.50) and (1.49) we see that

$$\langle P(t)\rangle=P_{\text{average}}=\frac{1}{\pi r}\int_{0}^{+\infty}S_U(\omega)d\omega. \tag{1.51}$$

In fact, the right-hand side of (1.51) is the sum of the elementary powers, each equal to $\frac{1}{\pi r}S_U(\omega)d\omega$, and the sum is taken over all the frequencies from $\omega=0$ to $\omega=+\infty$. So, up to a constant factor, $S_U(\omega)$ is the power within the frequency range from ω to $\omega+d\omega$. It should be stressed that if for a signal $x(t)$ $S_x(\omega_0)=0$, then the frequency ω_0 is absent in the signal $x(t)$.

Now let us obtain the spectral densities of some signals.

1. White noise $R_x(t)=C\delta(t)$, C is a constant and $\delta(t)$ is the Dirac delta function.

Let us recall some facts about the delta function. The delta-function is the function $\delta(t)$ such that

$$\delta(t)=\begin{cases}+\infty, t=0\\ 0, t\neq 0\end{cases},\quad \int_{-\infty}^{+\infty} dt\delta(t)=1. \tag{1.52}$$

The delta-function obeys the properties

$$\int_{-\infty}^{\infty} dx f(x)\delta(x-a)=f(a),\ \delta(t)=\frac{1}{2\pi}\int_{-\infty}^{\infty} d\omega e^{i\omega t}=\frac{1}{2\pi}\int_{-\infty}^{\infty} d\omega e^{-i\omega t},\quad \delta(t)=\delta(-t). \tag{1.53}$$

On the basis of (1.43) and (1.53) we have

$$S_x(\omega)=\int_{-\infty}^{+\infty} dtR_x(t)e^{-i\omega t}=C\int_{-\infty}^{+\infty} dte^{-i\omega t}\delta(t)=C. \tag{1.54}$$

So, the spectral density of white noise is a constant.

2. Constant signal $x(t)=x_0=\text{const}$. Obviously,

$$R_x(\tau)=\lim_{T\to\infty}\frac{1}{2T}\int_{-T}^{T} dtx(t+\tau)x(t)=x_0^2\lim_{T\to\infty}\frac{1}{2T}\int_{-T}^{T} dt=x_0^2=\text{const} \tag{1.55}$$

and from (1.43) and (1.53) one can obtain

$$S_x(\omega)\equiv\int_{-\infty}^{+\infty} dtR_x(t)e^{-i\omega t}=2\pi x_0^2\cdot\frac{1}{2\pi}\int_{-\infty}^{+\infty} dte^{-i\omega t}=2\pi x_0^2\delta(\omega). \tag{1.56}$$

So the spectral density of a constant signal is proportional to $\delta(\omega)$. Physically, such a result is evident because the only frequency within the constant signal is $\omega=0$.

3. Sine signal $x(t)=A\sin(\omega_1 t+\varphi)$. This signal is a periodical one, so we can do averaging only over one period during the calculation of the correlation function:

$$R_x(\tau)=\frac{1}{T}\int_0^T dt x(t+\tau)x(t)=\frac{A^2}{T}\int_0^T dt\sin(\omega_1 t+\omega_1\tau+\varphi)\sin(\omega_1 t+\varphi),\ T=\frac{2\pi}{\omega}. \quad (1.57)$$

On the basis of a well-known formula

$$\sin x\sin y=\frac{1}{2}(\cos(x-y)-\cos(x+y)) \quad (1.58)$$

we can rewrite (1.57) as

$$R_x(\tau)=\frac{A^2}{2T}\cos(\omega_1\tau)\int_0^T dt-\frac{A^2}{2T}\int_0^T dt\cos(2\omega_1 t+\omega_1\tau+2\varphi). \quad (1.59)$$

Obviously, the second summand on the right-rand side of (1.59) is equal to zero, so

$$R_x(\tau)=\frac{A^2}{2}\cos(\omega_1\tau). \quad (1.60)$$

From (1.43) and (1.53) we have

$$\begin{aligned} S_x(\omega)&\equiv\frac{A^2}{2}\int_{-\infty}^{+\infty}dt\cos(\omega_1 t)e^{-i\omega t}=\frac{A^2}{2}\int_{-\infty}^{+\infty}dt\frac{e^{-i\omega_1 t}+e^{i\omega_1 t}}{2}e^{-i\omega t}= \\ &=\frac{A^2\pi}{2}\frac{1}{2\pi}\int_{-\infty}^{+\infty}dt\left(e^{-i(\omega_1+\omega)t}+e^{i(\omega_1-\omega)t}\right)=\frac{A^2\pi}{2}\left(\delta(\omega+\omega_1)+\delta(\omega-\omega_1)\right). \end{aligned} \quad (1.61)$$

So $S_x(\omega)$ contains two delta-maximums at $\omega=\pm\omega_1$, for all the other frequencies the spectral density is equal to zero. Obviously, negative values of ω are unphysical, and all the power of the signal is "gathered" at the frequency ω_1, which is physically evident.

4. Periodic signal $x(t)$ with a period T_0. In such a case the signal can be expanded into a Fourier series

$$x(t)=A_0+\sum_{k=1}^{\infty}A_k\sin(\omega_k t+\varphi_k),\ \omega_k=\frac{2\pi k}{T_0}. \quad (1.62)$$

By a straightforward calculation on the basis of (1.58) it can be shown that all the harmonics of a Fourier series are independent, i.e.

$$\forall k \neq m \int_0^{T_0} \sin\left(\omega_k t + \varphi_k\right)\sin\left(\omega_m t + \varphi_m\right) = 0. \tag{1.63}$$

On the basis of (1.63) it is evident that

$$\begin{aligned} R_x(\tau) = \frac{1}{T_0}\int_0^{T_0} dt x(t+\tau)x(t) = A_0^2 + \sum_{k=1}^{\infty}\frac{A_k^2}{T_0}\int_0^{T_0} dt \sin\left(\omega_k t + \omega_k \tau + \varphi_k\right)\sin\left(\omega_k t + \varphi_k\right) = \\ = A_0^2 + \sum_{k=1}^{\infty}\frac{A_k^2}{2}\cos\left(\omega_k \tau\right) \end{aligned} \tag{1.64}$$

which leads to

$$S_x(\omega) = 2\pi A_0^2 \delta(\omega) + \pi\sum_{k=1}^{\infty}\frac{A_k^2}{2}\left(\delta\left(\omega+\omega_k\right) + \delta\left(\omega-\omega_k\right)\right). \tag{1.65}$$

So the signal contains only a zero frequency and frequencies that are multiples of $2\pi/T_0$.

It should be stressed that if a signal does not contain periodical components then its spectral density does not contain delta-maximums. If a signal contains periodical components (for example, a sum of white noise and the sine function) then its spectral density is a sum of a continuous function and delta-maximums at frequencies of the periodical harmonics.

1.4 Adaptive signal processing and digital adaptive filter

The general structure of the adaptive filter is shown in Fig. 1.1. The discrete input signal $x(k)$ is processed by a discrete filter, resulting in an output signal $y(k)$. This output signal is compared with the reference signal $d(k)$, the difference between them forms the error signal $e(k)$. The task of the adaptive filter is to minimize the reproduction error of the reference signal. To do this, the adaptation unit, after processing each sample, analyzes the error signal and additional data from the filter, using the results of this analysis to adjust the parameters (coefficients) of the filter.

Another adaptation option is possible, in which the reference signal is not used. This mode of operation is called **blind adaptation** or unsupervised learning. Of

course, in this case, some information is needed about the structure of the useful input signal (for example, knowledge of the type and parameters of the modulation used). It is obvious that blind adaptation is a more complex computational task than adaptation using the reference signal. We will not consider these algorithms.

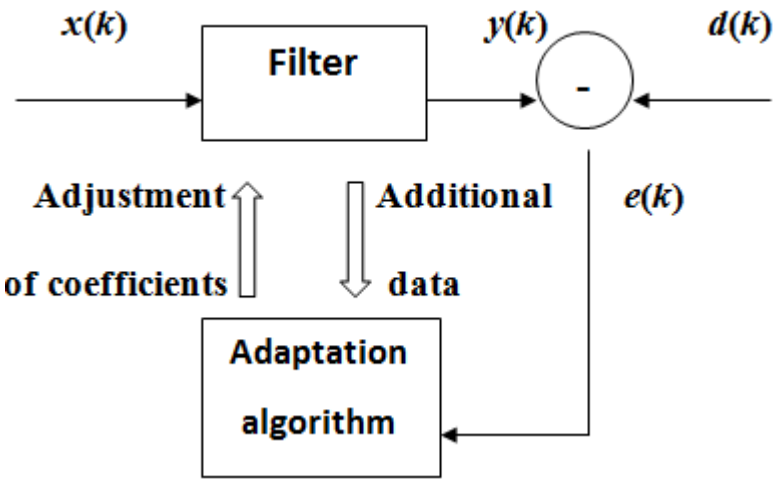


Figure 1.1 – General structure of the adaptive filter

It may seem that the algorithms using the reference signal are devoid of practical meaning, since the output signal must be known in advance. However, there are a number of practical problems in solving which the reference signal is available. It should be noted that in some cases, the useful signal is not the output signal of the filter, but the error signal, that is, the difference between the sample signal and the output signal of the adaptive filter.

The filter most commonly used in the structure shown in Fig. 1.1 is a non-recursive digital filter. One of the main advantages of this option is that the non-recursive filter is stable for any values of coefficients. However, it should be remembered that the adaptation algorithm in any case introduces feedback into the system, as a result of which the adaptive system as a whole may become unstable.

There are adaptive algorithms for recursive filters; however, when developing them, serious problems arise, primarily related to stability, therefore such filters are not widely used. Another class of adaptive systems is neural networks, which to some extent simulate the functioning of the nervous system of living organisms.

Next, we will consider three adaptive algorithms using the reference signal, often used in practice in various information processing systems.

1.5 Correlation matrix. Eigenvalues and eigenvectors of correlation matrix

1.5.1 Correlation matrix

Adaptive digital filter algorithms use in their equations the input signals of a linear adder (Fig. 1.2), measured at discrete points of time *k*. The results of these measurements form a vector of samples of input signals

$$X_N(k)=\left(x_1(k),x_2(k),...,x_N(k)\right)^t. \tag{1.66}$$

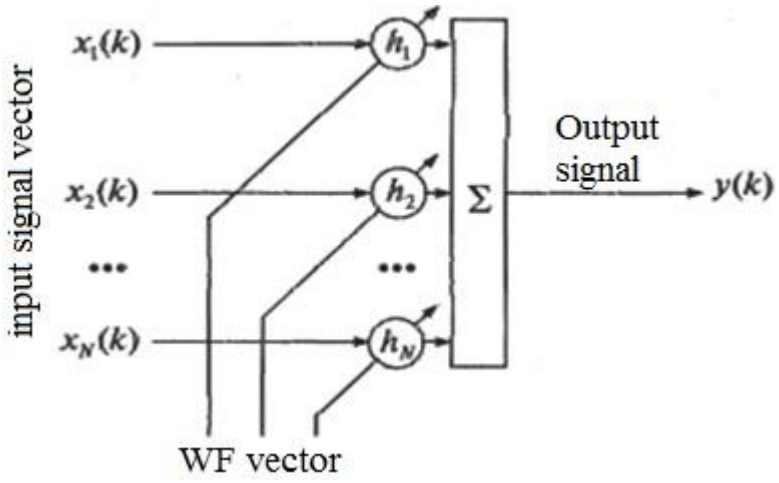


Figure 1.2 – Linear Adder

If the adaptive filter in its structure is a single-channel FIR filter, then the vector

$$X_N(k)=\left(x_1(k),x_2(k),...,x_N(k)\right)^t=\left(x(k),x(k-1),...,x(k-N+1)\right)^t \tag{1.67}$$

represents the delayed samples of the input signal of this filter, that is, a discrete stochastic process.

If the adaptive filter is a multichannel filter with one weighting factor (WF) in each channel, then

$$X_N(k)=\left(x_1(k),x_2(k),...,x_N(k)\right)^t=\left(x_1(k),x_2(k),...,x_M(k)\right)^t \tag{1.68}$$

where $N=M$.

If the adaptive filter is M-channel filter with a finite impulse response (FIR filter) with an unequal number of WF N_m in the channels, then

$$X_N(k)=\left(x_{N_1}^t(k),x_{N_2}^t(k),...,x_{N_M}^t(k)\right)^t \tag{1.69}$$

where

$$x_{N_m}(k)=\left(x_m(k),x_m(k-1),...,x_m(k-N_m+1)\right)^t,\ m=1,2,...,M;\ N=\sum_{m=1}^{M}N_m.$$

In this case, the signals may have the same or different (same within the channels) statistical characteristics.

Thus, in the general case, signals $x_n(k)$ (see Fig. 1.2) may have different statistical characteristics. The correlation matrix of these signals is defined as

$$R_N=\begin{pmatrix} E\left\{\left|x_1(k)\right|^2\right\} & E\left\{x_1(k)x_2^*(k)\right\} & \cdots & E\left\{x_1(k)x_N^*(k)\right\} \\ E\left\{x_2(k)x_1^*(k)\right\} & E\left\{\left|x_2(k)\right|^2\right\} & \cdots & E\left\{x_2(k)x_N^*(k)\right\} \\ \vdots & \vdots & \ddots & \vdots \\ E\left\{x_N(k)x_1^*(k)\right\} & E\left\{x_N(k)x_2^*(k)\right\} & \cdots & E\left\{\left|x_N(k)\right|^2\right\} \end{pmatrix} \tag{1.70}$$

where $E\{\}$ is the operation of averaging over the ensemble of realizations and * denotes the complex conjugation.

If the signals $x_n(k)$, n = 1, 2, ..., N, have the same statistical characteristics, then the correlation matrix (1.70) is defined as

$$R_N=\begin{pmatrix} R(0) & R(1) & R(2) & R(3) & \cdots & R(N-1) \\ R(-1) & R(0) & R(1) & R(2) & \cdots & R(N-2) \\ R(-2) & R(-1) & R(0) & R(1) & \cdots & R(N-3) \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ R(-N+2) & R(-N+3) & R(-N+4) & R(-N+5) & \cdots & R(1) \\ R(-N+1) & R(-N+2) & R(-N+3) & R(-N+4) & \cdots & R(0) \end{pmatrix} \tag{1.71}$$

In the matrix (1.71) $R(n-m)=E\{x_n(k)x_m^*(k)\}$, these are samples of the mutual correlation function, if the vector is determined by equation (1.68), and $R(n-m)=E\{x(k-n+1)x^*(k-m+1)\}$ are the samples of the autocorrelation function, if the vector $X_N(k)$ is defined by equation (1.67). The diagonal elements $R(0)$ of the matrix (1.71) are the same. They are equal to the rms values of the input signals of the linear adder, that is, they are real positive numbers. The remaining elements of $R(1)$, ..., $R(N-1)$, $R(-1)$, ..., $R(-N+1)$ are real or complex depending on whether signals of the linear adder are real or complex. For these elements, the relation is always valid $|R(n-m)| \le R(0)$, since they represent the values of the side lobes of the auto or mutual correlation function, which in absolute value do not exceed the values of the main lobe $R(0)$ of this function. The correlation matrix has a number of properties, the main ones without proof are given below.

Property 1. The correlation matrix is Hermitian, that is,

$$R_N = R_N^H .$$

A square matrix is called *Hermitian* (or *self-adjoint*) if each of its elements is complexly conjugated to an element that is symmetric to this relative to the main diagonal; in other words, the matrix A is Hermitian, if

$$a_{ik} = a_{ki}^* .$$

Property 2. The correlation matrix of a stationary discrete stochastic process is a *Toeplitz* one. A Toeplitz matrix is a matrix, in which the same elements on the main diagonal and the elements on each of the diagonals parallel to the main one are also the same.

Property 3. The correlation matrix is positively semidefinite, that is, the condition $h_N^H R_N h_N \ge 0$ is satisfied. If $h_N^H R_N h_N > 0$, then the matrix R_N is positive definite.

In practice, the matrix R_N is positively defined except for the rare cases when the signals in the vector (1.66) are linearly dependent. Positive definiteness means that the matrix is nonsingular, that is, its determinant $\det(R_N) > 0$ which means that there exists an inverse matrix because

$$R_N^{-1} = \frac{1}{\det(R_N)} \hat{R}_N = \frac{1}{\det(R_N)} \begin{pmatrix} \hat{R}_{11} & \hat{R}_{21} & \cdots & \hat{R}_{N1} \\ \hat{R}_{21} & \hat{R}_{22} & \cdots & \hat{R}_{N2} \\ \vdots & \vdots & \ddots & \vdots \\ \hat{R}_{N1} & \hat{R}_{N2} & \cdots & \hat{R}_{NN} \end{pmatrix} \tag{1.72}$$

where $\hat{R}_N$ is the adjoint matrix made up of algebraic complements of the matrix R_N. To obtain an adjoint matrix, you need to transpose the original R_N matrix, and then replace all elements of the transposed matrix with algebraic adjuncts.

Property 4. The correlation matrix of signals of a multichannel filter can be represented as a cell matrix.

From equations (1.69) and (1.70) it follows that the correlation matrix of signals of a multichannel filter is defined as

$$R_N = \begin{pmatrix} R_{N_1} & R_{N_1 N_2} & \cdots & R_{N_1 N_M} \\ R_{N_2 N_1} & R_{N_2} & \cdots & R_{N_2 N_M} \\ \vdots & \vdots & \ddots & \vdots \\ R_{N_M N_1} & R_{N_M N_2} & \cdots & R_{N_M} \end{pmatrix} \tag{1.73}$$

where $R_{N_j} = E\left\{x_{N_j}(k) x_{N_j}^H(k)\right\}$, $R_{N_i N_j} = E\left\{x_{N_i}(k) x_{N_j}^H(k)\right\}$.

From (1.73) it can be seen that on the main diagonal of the correlation matrix of a multichannel filter are located the square correlation matrices of the input signals of individual channels, and the remaining elements are matrices of mutual correlation of signals between the channels. The matrix structure (1.73) indicates that the design of adaptation algorithms should take into account the relationship between the signals of the filter channels.

1.5.2 Eigen-values and eigenvectors of the correlation matrices

One of the important properties of the correlation matrix is that this matrix can be decomposed into eigen-values and the corresponding eigenvectors.

The number λ is called the eigen-value of the matrix R_N, if the condition

$$R_N q_N = \lambda q_N \tag{1.74}$$

is satisfied where $q_N \neq 0$ is the eigenvector corresponding to this eigen-value. Equation (1.74) means that the Hermitian matrix R_N performs linear transformation of the vector q_N into a vector λq_N, that is, into a vector that coincides in direction with the original non-transformed vector, since it will be shown later that the eigen-values of the matrix are real and non-negative.

Equation (1.74) can be represented as

$$R_N q_N - \lambda q_N = \left(R_N - \lambda I_N\right) q_N \tag{1.75}$$

Equation (1.75) is a homogeneous linear equation with unknown q_N. It has a non-trivial solution $q_N \neq 0$ only if

$$\det\left(R_N - \lambda I_N\right) = 0 \tag{1.76}$$

Equation (1.76) is called the characteristic equation of the matrix R_N and is an N_{th}-degree polynomial

$$p(\lambda) = a_N \lambda^N + a_{N-1}\lambda^{N-1} + ... + a_1 \lambda + a_0 . \tag{1.77}$$

Equation (1.76) has N roots (not necessarily different), λ_1, $\lambda_2, \ldots, \lambda_N$ called eigen-values of the matrix R_N.

Each vector $q_{N,n} \neq 0$ where $n = 1, 2, ..., N$, satisfying the equation

$$R_N q_{N,n} = \lambda_n q_{N,n} . \tag{1.78}$$

is called the eigenvector corresponding to the eigen-value λ_n.

Each eigenvector corresponds to only one eigen-value, while one eigen-value corresponds to an infinite set of eigenvectors.

The eigen-values λ_n and eigenvectors $q_{N,n}$ of the correlation matrix R_N of a stationary discrete stochastic process are characterized by a number of properties, which we present without proof.

Property 1. If λ_1, λ_2, …, λ_N are the eigen-values of the correlation matrix R_N, then for any integer $k>0$, the eigen-values of the matrix $R_N^k = \underbrace{R_N R_N ... R_N}_{k}$ are equal to λ_1^k, λ_2^k, …, λ_N^k.

Property 2. The non-zero eigenvectors $q_{N,1}$, $q_{N,2}$, …, $q_{N,N}$ of the correlation matrix R_N, corresponding to different eigen-values λ_1, λ_2, …, λ_N, are linearly independent. Vectors $q_{N,n}$ are linearly dependent if there exist $a_i \neq 0$, $i=\overline{1,N}$ such that

$$a_1 q_{N,1} + a_2 q_{N,2} + ... + a_N q_{N,N} = 0. \qquad (1.79)$$

If equality (1.79) holds only for all $a_i = 0$, then the vectors $q_{N,i}$ are linearly independent.

Property 3. The eigen-values λ_1, λ_2, …, λ_N of the correlation matrix R_N are non-negative and real.

Property 4. The eigenvectors $q_{N,1}$, $q_{N,2}$, …, $q_{N,N}$ of the correlation matrix R_N, corresponding to different eigen-values λ_1, λ_2, …, λ_N, are orthogonal to each other. Orthogonality means that

$$q_{N,n}^H q_{N,m} = 0 \text{ if } n \neq m. \qquad (1.80)$$

As noted earlier, each eigen-value λ_n corresponds to an infinite set of vectors $aq_{N,n}$ with $a \neq 0$. Therefore, without loss of generality, in what follows we will consider only normalized eigenvectors, that is, those for which realized the following condition holds:

$$q_{N,n}^H q_{N,m} = \sum_{i=1}^{m} q_{i,n}^* q_{i,m}^* = \delta_{mn} = \begin{cases} 1, m = n \\ 0, m \neq n \end{cases} \tag{1.81}$$

for all n = 1,2, ..., N and m = 1,2, ..., N. Vectors that satisfy condition (1.81) are called orthonormalized.

Property 5. If the eigenvectors $q_{N,1}$, $q_{N,2}$, …, $q_{N,N}$ of the correlation matrix R_N, corresponding to different eigen-values, are represented by the matrix $Q_N = \left(q_{N,1}, q_{N,2}, q_{N,3}, ..., q_{N,N}\right)$ of orthonormalized vectors (1.81), and the eigen-values are represented by the diagonal matrix

$$\Lambda_N = \operatorname{diag}\left(\lambda_1, \lambda_2, ..., \lambda_N\right) = \begin{pmatrix} \lambda_1 & 0 & \cdots & 0 \\ 0 & \lambda_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \lambda_N \end{pmatrix}, \tag{1.82}$$

then the matrix R_N can be reduced to a diagonal form $Q_N^H R_N Q_N = \Lambda_N$.

The condition (1.81) means that the vectors $q_{N,1}$, $q_{N,2}$, …, $q_{N,N}$ form an orthonormalized set. Using the definition of a matrix Q_N, the system of equations (1.78) can be represented as a matrix equation

$$R_N Q_N = Q_N \Lambda_N . \tag{1.83}$$

According to (1.81) we have

$$Q_N^H Q_N = \begin{pmatrix} q_{N,1}^H q_{N,1} & q_{N,1}^H q_{N,2} & \cdots & q_{N,1}^H q_{N,N} \\ q_{N,2}^H q_{N,1} & q_{N,2}^H q_{N,2} & \cdots & q_{N,2}^H q_{N,N} \\ \vdots & \vdots & \ddots & \vdots \\ q_{N,N}^H q_{N,1} & q_{N,N}^H q_{N,2} & \cdots & q_{N,N}^H q_{N,N} \end{pmatrix} = \begin{pmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 \end{pmatrix} = I_N . \tag{1.84}$$

from where

$$Q_N^H = Q_N^{-1} . \tag{1.85}$$

From (1.85) it follows that $Q_N Q_N^H = Q_N Q_N^{-1} = I_N$. A complex matrix with property (1.85) is called unitary, and the analogous real matrix is orthonormalized.

If equation (1.83) is multiplied to the right of the matrix Q_N^H, i.e., $R_N Q_N Q_N^H = Q_N \Lambda_N Q_N^H$ then using (1.84) we get

$$R_N = Q_N \Lambda_N Q_N^H = \sum_{n=1}^{N} \lambda_n q_{N,n} q_{N,n}^H . \tag{1.86}$$

Equation (1.86) is the spectral decomposition of the matrix R_N or the decomposition on eigen-values and eigenvectors.

Similarly, using the equality (1.78) and property 1 as $R_N^k Q_N = Q_N \Lambda_N^k$, it can be shown that

$$R_N^k Q_N Q_N^H = Q_N \Lambda_N^k Q_N^H \Rightarrow R_N^k = \sum_{n=1}^{N} \lambda_n^k q_{N,n} q_{N,n}^H . \tag{1.87}$$

From (1.86) it follows that the average value of the square of the modulus of the output signal of the adaptive filter is defined as

$$\begin{aligned} &E\left\{\left|y(k)\right|^2\right\} = E\left\{y(k)y^*(k)\right\} = E\left\{h_N^H x_N x_N^H h_N\right\} = h_N^H E\left\{x_N x_N^H\right\} h_N = \\ &= h_N^H R_N h_N = h_N^H \sum_{n=1}^{N} \lambda_n q_{N,n} q_{N,n}^H h_N = \sum_{n=1}^{N} \lambda_n (h_N^H q_{N,n})(q_{N,n}^H h_N) = \sum_{n=1}^{N} \lambda_n \left|h_N^H q_{N,n}\right|^2 , \end{aligned} \tag{1.88}$$

that is, it is equal to the sum of the squares of the modules of the scalar products of the WC filter vector and the eigenvectors of the correlation matrix of its signals, weighted by the eigen-values.

Using equation (1.85) and the property of invertible square matrices

$$\left(A_N B_N\right)^{-1} = B_N^{-1} A_N^{-1}, \tag{1.89}$$

from (1.86) we can define the inverse matrix R_N^{-1} as

$$R_N^{-1} = \left(Q_N \Lambda_N Q_N^H\right)^{-1} = Q_N^H \Lambda_N^{-1} Q_N = \sum_{n=1}^{N} \lambda_n^{-1} q_{N,n} q_{N,n}^H . \tag{1.90}$$

Property 6. The sum of the eigen-values λ_1, λ_2, …, λ_N of the correlation matrix R_N is equal to the trace of the matrix.

The trace of the square matrix A_N is the sum of its diagonal elements:

$$\mathrm{Tr}\left(A_N\right) = \sum_{n=1}^{N} a_{nn} \, .$$

Property 7. The product of the eigen-values λ_1, λ_2, …, λ_N of the correlation matrix R_N is equal to the determinant of this matrix:

$$\det\left(R_N\right) = \det\left(\Lambda_N\right) = \prod_{i=1}^{N} \lambda_i \, . \qquad (1.91)$$

The information in the first chapter should help the reader gain a deeper understanding of the subsequent chapters.

Chapter 2. Analog methods of adaptive filtration

2.1. Linear stationary filters

In what follows we understand a **filter** as a physical device or a program that enhances the useful signal from a mixture of the useful signal and a noise or makes a prediction of future values of the useful signal.

Suppose that for an input signal $x_1(t)$ the filter generates an output signal $y_1(t)$, and for an input signal $x_2(t)$ it generates an output signal $y_2(t)$. Let the input signal be a linear combination of the signals $x_1(t)$ and $x_2(t)$: $x(t) = Ax_1(t) + Bx_2(t)$ where A, B are some constants. Then the filter is **linear** if its output signal is a linear combination of the corresponding signals $y_1(t)$ and $y_2(t)$ with the same constants A, B: $y(t) = Ay_1(t) + By_2(t)$.

Suppose that the input signal lags by some time. Then the filter is **stationary** if its output signal lags by the same time. In other words, suppose that for an input signal $x(t)$ the filter generates an output signal $y(t)$. Let the input signal be equal to $x(t - \Delta t)$. Then the filter is **stationary** if it generates the output signal $y(t - \Delta t)$.

For simplicity, in what follows we consider only analog signals that are defined on the whole time axis from $-\infty$ to $+\infty$. Let the input and output signals of a linear stationary filter be $x(t)$ and $y(t)$, respectively. Then the **weight function** of that filter is the function $h(t)$ such that

$$y(t) = \int_{-\infty}^{+\infty} d\tau h(\tau) x(t - \tau). \qquad (2.1)$$

Let us show that a filter described by (2.1) is a linear stationary one. Let an output signal $y_1(t)$ correspond to an input signal $x_1(t)$, and an output signal $y_2(t)$ correspond to an input signal $x_2(t)$. According to (2.1)

$$y_1(t)=\int_{-\infty}^{+\infty} d\tau h(\tau)x_1(t-\tau),\quad y_2(t)=\int_{-\infty}^{+\infty} d\tau h(\tau)x_2(t-\tau). \tag{2.2}$$

Let the input signal be equal to

$$X(t)=Ax_1(t)+Bx_2(t) \tag{2.3}$$

where A, B are some constants. Let us calculate the output signal $Y(t)$ that corresponds to the input signal $X(t)$. According to (2.1) – (2.3), we have

$$\begin{aligned} Y(t)=\int_{-\infty}^{+\infty} d\tau h(\tau)X(t-\tau)=A\int_{-\infty}^{+\infty} d\tau h(\tau)x_1(t-\tau)+B\int_{-\infty}^{+\infty} d\tau h(\tau)x_2(t-\tau)= \\ =Ay_1(t)+By_2(t), \end{aligned} \tag{2.4}$$

so it is proved that the filter is linear.

From (2.1) it is clear that

$$y(t-\Delta t)=\int_{-\infty}^{+\infty} d\tau h(\tau)x(t-\Delta t-\tau), \tag{2.5}$$

so the filter is stationary.

The weight function of a linear stationary filter obeys the property

$$h(t<0)=0. \tag{2.6}$$

This property is a consequence of the **causality principle,** which states that a change in the output function must not occur earlier than a change in the input function. Let us consider a simple example where the input function is

$$x(t)=\delta(t-t_0), \tag{2.7}$$

so the input function is nonzero only at the instant $t=t_0$, and on the basis of the causality principle we have

$$y(t<t_0)=0. \tag{2.8}$$

From (2.1), (2.7) and (1.53), we obtain

$$y(t)=\int_{-\infty}^{+\infty} d\tau h(\tau)\delta(t-t_0-\tau)=h(t-t_0) \tag{2.9}$$

which with account for (2.8) leads to:

$$h(t-t_0)=0 \text{ if } t<t_0. \tag{2.10}$$

From (2.10) it is clear that the weight function of a negative argument is indeed equal to zero.

2.2. The Kolmogorov-Wiener filter

Let the input signal $x(t)$ be a sum of a useful signal $s(t)$ and a noise $n(t)$:

$$x(t)=n(t)+s(t), \tag{2.11}$$

$s(t)$ and $n(t)$ are stationary random processes. The cross-correlation function $R_{sn}(t)$ of the processes $s(t)$ and $n(t)$ is considered to be stationary. For simplicity, we consider only analog signals for times $t\in(-\infty,+\infty)$, and all the processes are supposed to be ergodic. *It is assumed that the useful signal and the noise are not known, but the cross-correlation function $R_{sx}(t)$ of the processes $s(t)$ and $x(t)$ and the correlation function $R_x(t)$ of the process $x(t)$ are known.*

The Kolmogorov–Wiener filter is a linear stationary filter which can both enhance the useful signal from a mixture of the useful signal and a noise and predict future values of the useful signal. ***The aim of the filter is to generate the output signal $y(t)$ closest to the useful signal $s(t+\eta)$ where $\eta\geq 0$ is a specified parameter***. ***The word "closest" should be understood as "having the smallest mean-square deviation".*** The physical meaning of the parameter η is the time interval for which the prediction is made.

As mentioned in the previous section, the output and the input signals of a linear stationary filter are related as

$$y(t)=\int_{-\infty}^{+\infty} d\tau h(\tau)x(t-\tau). \tag{2.12}$$

As mentioned above, the aim of the filter is to minimize the mean-square deviation

$$\xi=\left\langle \left(s(t+\eta)-y(t)\right)^2\right\rangle_t \to \min. \tag{2.13}$$

So our aim is to find an equation for the weight function $h(\tau)$ *on the basis of (2.12) and (2.13).*

Obviously,

$$\xi=\left\langle \left(s(t+\eta)-y(t)\right)^2\right\rangle_t=\left\langle s^2(t+\eta)\right\rangle_t-2\left\langle s(t+\eta)y(t)\right\rangle_t+\left\langle y^2(t)\right\rangle_t \tag{2.14}$$

and with account for (2.14) and (2.12) we obtain

$$\begin{gathered}\xi=\xi_1+\xi_2+\xi_3,\ \ \xi_1=\left\langle s^2(t+\eta)\right\rangle_t,\ \ \xi_2=-2\left\langle s(t+\eta)\int_{-\infty}^{+\infty} d\tau h(\tau)x(t-\tau)\right\rangle_t,\\ \xi_3=\left\langle \int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} d\tau d\tau' h(\tau)h(\tau')x(t-\tau)x(t-\tau')\right\rangle_t.\end{gathered} \tag{2.15}$$

Now we should calculate ξ_1, ξ_2 and ξ_3. The processes are considered to be ergodic, so the assembly average and the time average coincide. From the definition (1.10) and the property (1.14) we have

$$\begin{gathered}\xi_1=\left\langle s^2(t+\eta)\right\rangle_t=\int_{-\infty}^{\infty} dss^2\omega_1(s,t+\eta)=\int_{-\infty}^{\infty} dss^2\omega_1(s,t+\eta-\eta)=\\ =\int_{-\infty}^{\infty} dss^2\omega_1(s,t)=\left\langle s^2(t)\right\rangle_t.\end{gathered} \tag{2.16}$$

At the same time we have that the correlation function $R_s(\tau)$ is

$$R_s(\tau)=\left\langle s(t)s(t+\tau)\right\rangle_t \Rightarrow R_s(0)=\left\langle s^2(t)\right\rangle_t. \tag{2.17}$$

So on the basis of (2.16) and (2.17) we have

$$\xi_1=R_s(0). \tag{2.18}$$

As for ξ_2, on the basis of (2.15) we have

$$\xi_2 = -2\left\langle s(t+\eta)\int_{-\infty}^{+\infty} d\tau h(\tau)x(t-\tau)\right\rangle_t = -2\left\langle \int_{-\infty}^{+\infty} d\tau h(\tau)s(t+\eta)x(t-\tau)\right\rangle_t = \\ = -2\int_{-\infty}^{+\infty} d\tau h(\tau)\left\langle s(t+\eta)x(t-\tau)\right\rangle_t, \tag{2.19}$$

the last equality in (2.19) is valid because the averaging is made over the parameter t and $h(\tau)$ is independent of t.

From (1.28) and (1.30) we have

$$\left\langle s(t+\eta)x(t-\tau)\right\rangle_t = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} dsdxsx\omega(s,t+\eta;x,t-\tau) = \\ = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} dsdxsx\omega(s,t+\eta+\tau;x,t) = \left\langle s(t+\eta+\tau)x(t)\right\rangle_t = R_{sx}(\eta+\tau). \tag{2.20}$$

So from (2.20) and (2.19) it can be seen that

$$\xi_2 = -2\int_{-\infty}^{+\infty} d\tau h(\tau)R_{sx}(\tau+\eta). \tag{2.21}$$

Similarly, on the basis of (2.15) we have

$$\xi_3 = \left\langle \int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} d\tau d\tau' h(\tau)h(\tau')x(t-\tau)x(t-\tau')\right\rangle_t = \\ = \int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} d\tau d\tau' h(\tau)h(\tau')\left\langle x(t-\tau)x(t-\tau')\right\rangle_t. \tag{2.22}$$

From (1.11), (1.14) and (1.23) we can obtain

$$\left\langle x(t-\tau)x(t-\tau')\right\rangle_t = \int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} dx_1dx_2x_1x_2\omega_2(x_1,t-\tau;x_2,t-\tau') = \\ = \int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} dx_1dx_2x_1x_2\omega_2(x_1,t;x_2,t-\tau'+\tau) = \left\langle x(t)x(t+\tau-\tau')\right\rangle_t = R_x(\tau-\tau'), \tag{2.23}$$

so on the basis of (2.22) and (2.23) it can be seen that

$$\xi_3 = \int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} d\tau d\tau' h(\tau)h(\tau')R_x(\tau-\tau'). \tag{2.24}$$

Finally, from (2.15), (2.18), (2.20) and (2.23) we have

$$\xi = \xi\left[h(\tau)\right] = R_s(0) - 2\int_{-\infty}^{+\infty} d\tau h(\tau) R_{sx}(\tau+\eta) + \\ + \int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} d\tau d\tau' h(\tau) h(\tau') R_x(\tau-\tau') \qquad (2.25)$$

As can be seen from (2.25), the mean-square deviation ξ is a functional of the filter weight function $h(\tau)$. Our aim is to minimize this functional, i.e. to find the function $h(\tau)$ for which the mean-square deviation ξ takes the minimal value. The idea of this minimization is as follows. Let $h(\tau)$ be the weight function that minimizes the functional (2.25). If we take another weight function in the form $h(\tau)+\varepsilon g(\tau)$ where ε is an arbitrary number and $g(\tau)$ is an arbitrary function, then the following inequality must hold:

$$\forall \varepsilon,\ \forall g(\tau)\quad \xi\left[h(\tau)\right] \le \xi\left[h(\tau)+\varepsilon g(\tau)\right], \qquad (2.26)$$

because the function $h(\tau)$ minimizes the functional (2.25).

On the basis of (2.25) we have

$$\xi\left[h(\tau)\right] - \xi\left[h(\tau)+\varepsilon g(\tau)\right] = \\ = R_s(0) - 2\int_{-\infty}^{+\infty} d\tau h(\tau) R_{sx}(\tau+\eta) + \int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} d\tau d\tau' h(\tau) h(\tau') R_x(\tau-\tau') - \\ -\left(R_s(0) - 2\int_{-\infty}^{+\infty} d\tau \left(h(\tau)+\varepsilon g(\tau)\right) R_{sx}(\tau+\eta) + \right. \\ \left. + \int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} d\tau d\tau' \left(h(\tau)+\varepsilon g(\tau)\right)\left(h(\tau')+\varepsilon g(\tau')\right) R_x(\tau-\tau') \right). \qquad (2.27)$$

After removing brackets in (2.27) we obtain

$$\xi\left[h(\tau)\right] - \xi\left[h(\tau)+\varepsilon g(\tau)\right] = \\ = \varepsilon\left(-2\int_{-\infty}^{+\infty} d\tau g(\tau) R_{sx}(\tau+\eta) + \int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} d\tau d\tau' h(\tau) g(\tau') R_x(\tau-\tau') + \right. \qquad (2.28)$$

$$+\int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} d\tau d\tau' g(\tau)h(\tau')R_x(\tau-\tau')\Big) - \varepsilon^2 \int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} d\tau d\tau' g(\tau)g(\tau')R_x(\tau-\tau').$$

$R_x(\tau)$ is an even function, so

$$R_x(\tau-\tau') = R_x(\tau'-\tau), \tag{2.29}$$

which leads to

$$\int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} d\tau d\tau' h(\tau)g(\tau')R_x(\tau-\tau') = \int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} d\tau d\tau' g(\tau)h(\tau')R_x(\tau-\tau'), \tag{2.30}$$

so (2.28) takes the form

$$\begin{gathered}\xi[h(\tau)] - \xi[h(\tau)+\varepsilon g(\tau)] = \\ = -2\varepsilon\left(\int_{-\infty}^{+\infty} d\tau g(\tau)R_{sx}(\tau+\eta) - \int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} d\tau d\tau' h(\tau)g(\tau')R_x(\tau-\tau')\right) - \\ -\varepsilon^2 \int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} d\tau d\tau' g(\tau)g(\tau')R_x(\tau-\tau'),\end{gathered} \tag{2.31}$$

and from (2.31) and (2.26) it can be seen that the following inequality must hold:

$$\begin{gathered}\forall\varepsilon,\ \forall g(\tau)\ \ 2\varepsilon\left(\int_{-\infty}^{+\infty} d\tau g(\tau)R_{sx}(\tau+\eta) - \int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} d\tau d\tau' h(\tau)g(\tau')R_x(\tau-\tau')\right) + \\ +\varepsilon^2 \int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} d\tau d\tau' g(\tau)g(\tau')R_x(\tau-\tau') \geq 0.\end{gathered} \tag{2.32}$$

Let us consider the term multiplying ε^2. From (2.23) we have

$$\begin{gathered}\int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} d\tau d\tau' g(\tau)g(\tau')R_x(\tau-\tau') = \int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} d\tau d\tau' g(\tau)g(\tau')\langle x(t-\tau)x(t-\tau')\rangle_t = \\ = \left\langle \int_{-\infty}^{+\infty} d\tau g(\tau)x(t-\tau)\int_{-\infty}^{+\infty} d\tau' g(\tau')x(t-\tau')\right\rangle_t = \\ = \left\langle \left(\int_{-\infty}^{+\infty} d\tau g(\tau)x(t-\tau)\right)^2\right\rangle_t \geq 0\end{gathered} \tag{2.33}$$

so $\forall\varepsilon$, $\forall g(\tau)$ the term in (2.32) which contains ε^2 is non-negative.

So, the term in (2.32) which contains ε must also be non-negative for any number ε and for any function $g(\tau)$. The number ε can be either positive or negative, so such a requirement is valid only if the sum in parentheses multiplying ε is zero:

$$\forall g(\tau) \quad \int_{-\infty}^{+\infty} d\tau g(\tau) R_{sx}(\tau+\eta) - \int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} d\tau d\tau' h(\tau) g(\tau') R_x(\tau-\tau') = 0 \tag{2.34}$$

which leads to

$$\forall g(\tau) \quad \int_{-\infty}^{+\infty} d\tau' g(\tau') \left(R_{sx}(\tau'+\eta) - \int_{-\infty}^{+\infty} d\tau h(\tau) R_x(\tau-\tau') \right) = 0. \tag{2.35}$$

The equality (2.35) must be valid for any function $g(\tau)$, so the sum in parentheses must vanish:

$$R_{sx}(\tau'+\eta) - \int_{-\infty}^{+\infty} d\tau h(\tau) R_x(\tau-\tau') = 0. \tag{2.36}$$

On the basis of (2.36) and (2.6) we finally obtain

$$R_{sx}(\tau'+\eta) = \int_{0}^{+\infty} d\tau h(\tau) R_x(\tau-\tau'). \tag{2.37}$$

So the weight function that minimizes the functional (2.25) obeys the obtained integral equation (2.37). *Equation (2.37) is called the Wiener–Hopf equation.* In other words, ***the Kolmogorov-Wiener filter weight function*** $h(\tau)$ ***obeys the Wiener–Hopf equation (2.37).***

Let us recall that in (2.37) $R_{sx}(\tau)$ is the cross-correlation function between the input and the useful signal and $R_x(\tau)$ is the correlation function of the input signal.

The output signal of the Kolmogorov–Wiener filter can be obtained by the following algorithm. First of all, the filter weight function should be obtained on the basis of the Wiener–Hopf equation. Then the filter output can be found on the basis of the obtained weight function and the expression (2.12).

It should be stressed that up to this point only analog signals for times from $-\infty$ to $+\infty$ were considered. In the case of a discrete signal defined only for $t = 0,1,2,...,T$, the integral in (2.37) should be replaced with a sum and the infinite limits of integration should be replaced with finite bounds of summation:

$$R_{sx}(\tau' + \eta) = \sum_{\tau=0}^{T} h(\tau) R_x(\tau - \tau'), \quad y(t) = \sum_{\tau=0}^{T} h(\tau) x(t - \tau), \tag{2.38}$$

so in such a case we deal with a set of linear equations for $h(\tau)$, $\tau = 0,1,2,..,T$, and the output signal is the corresponding sum.

2.3. Scalar Kalman–Bucy filter

In fact, the Kalman–Bucy filter is similar to the Kalman one, but in contrast to the latter, the former is applied to continuous systems rather than to discrete ones. The scalar discrete Kalman filter is described in detail with an illustrative example in Sec. 3.6, so here we restrict ourselves to a general description of the Kalman–Bucy filter. For simplicity, we will consider only a scalar Kalman–Bucy filter, i.e. we will consider the case where the system under consideration is completely described by one parameter.

Let us have a system described by one continuously variable parameter x, and on the basis of some physical considerations the "ideal" law of the time evolution of this parameter can be written as

$$\frac{dx(t)}{dt} = a(t)x(t) + b(t)u(t) \tag{2.39}$$

where $u(t)$ is the known quantity that governs the system evolution; $a(t)$ and $b(t)$ are some known functions. The parameter $x(t)$ is measured by a device whose readings are $z(t)$. Let $\eta(t)$ be a random stationary process whose physical meaning is the device error. Let $\xi(t)$ be a random stationary process that describes the random character of the system evolution. So we have

$$\frac{dx(t)}{dt} = a(t)x(t) + b(t)u(t) + \xi(t), \quad z(t) = c(t)x(t) + \eta(t) \tag{2.40}$$

where the known function $c(t)$ ant its physical meaning is the device gain coefficient. In fact, the filter input is the function $z(t)$ and ***the aim of the filter is to generate the output function*** $x^{\text{opt}}(t)$ ***closest to*** $x(t)$***. The word "closest" should be understood as "having the smallest mean-square deviation".***

The filter error is introduced as

$$e(t) = x(t) - x^{\text{opt}}(t) \tag{2.41}$$

and the aim of the filter is to minimize the quantity

$$\forall t \ \left\langle e^2(t) \right\rangle \to \min . \tag{2.42}$$

The following assumptions are made:

1. The correlation functions of the processes $\eta(t)$ and $\xi(t)$ have the form

$$R_\xi(t,\tau) = Q(t)\delta(t-\tau), \ R_\eta(t,\tau) = R(t)\delta(t-\tau) \tag{2.43}$$

where $Q(t)$ and $R(t)$ are known functions. In fact, the processes $\eta(t)$ and $\xi(t)$ are assumed to behave as white noise. It should also be noticed that these processes may not be stationary.

2. The average values of $\eta(t)$ and $\xi(t)$ are zeros:

$$\forall t \ \left\langle \xi(t) \right\rangle = 0, \left\langle \eta(t) \right\rangle = 0. \tag{2.44}$$

3. The processes $\eta(t)$, $\xi(t)$ and $e(t)$ are independent ones:

$$\begin{gathered} \forall t_1, t_2 \ \left\langle \xi(t_1)\eta(t_2) \right\rangle = \left\langle \xi(t_1) \right\rangle \left\langle \eta(t_2) \right\rangle = 0, \left\langle \xi(t_1)e(t_2) \right\rangle = \left\langle \xi(t_1) \right\rangle \left\langle e(t_2) \right\rangle = 0, \\ \left\langle e(t_1)\eta(t_2) \right\rangle = \left\langle e(t_1) \right\rangle \left\langle \eta(t_2) \right\rangle = 0. \end{gathered} \tag{2.45}$$

The time derivative of $x^{\text{opt}}(t)$ is sought in the form

$$\frac{dx^{\text{opt}}(t)}{dt} = K(t)\left\{ z(t) - c(t)x^{\text{opt}}(t) \right\} + a(t)x^{\text{opt}}(t) + b(t)u(t) \tag{2.46}$$

where the unknown function $K(t)$ is the Kalman coefficient, which is sought on the basis of (2.42). On the basis of (2.40), (2.41) and (2.46) we can obtain

$$\frac{de(t)}{dt} = \frac{dx(t)}{dt} - \frac{dx^{\text{opt}}(t)}{dt} = \{a(t) - K(t)c(t)\}e(t) + \xi(t) - K(t)\eta(t). \tag{2.47}$$

Let us take the integral of both sides of (2.47):

$$\begin{aligned} \int_t^{t+dt} d\tau \frac{de(\tau)}{d\tau} = \int_t^{t+dt} d\tau \{a(\tau) - K(\tau)c(\tau)\}e(\tau) + \\ + \int_t^{t+dt} d\tau \xi(\tau) - \int_t^{t+dt} d\tau K(\tau)\eta(\tau), \end{aligned} \tag{2.48}$$

here $dt > 0$ is a very small time increment.

Obviously,

$$\int_t^{t+dt} d\tau \frac{de(\tau)}{d\tau} = e(t+dt) - e(t). \tag{2.49}$$

Let us consider the expression $\int_t^{t+dt} d\tau \varphi(\tau)$. The function $\varphi(\tau)$ may be expressed in terms of a Taylor polynomial in the vicinity of the point t:

$$\varphi(\tau) = \varphi(t) + \left.\frac{d\varphi(\tau)}{d\tau}\right|_{\tau=t} (t-\tau) + o(t-\tau). \tag{2.50}$$

From (2.49) and (2.50) we obtain

$$\begin{aligned} &\int_t^{t+dt} \varphi(\tau)d\tau = \varphi(t)\int_t^{t+dt} d\tau + \left.\frac{d\varphi(\tau)}{d\tau}\right|_{\tau=t} \int_t^{t+dt} d\tau(t-\tau) + \int_t^{t+dt} d\tau o(t-\tau) = \\ &= \left\{\begin{array}{l} \tau' = \tau - t, d\tau' = d\tau, \\ \tau = t \to \tau' = 0, \\ \tau = t + dt \to \tau' = dt \end{array}\right\} = \varphi(t)dt - \left.\frac{d\varphi(\tau)}{d\tau}\right|_{\tau=t} \int_0^{dt} d\tau'\tau' + o(dt^2) = \\ &= \varphi(t)dt - \left.\frac{d\varphi(\tau)}{d\tau}\right|_{\tau=t} \frac{dt^2}{2} + o(dt^2) = \varphi(t)dt + O(dt^2), \end{aligned} \tag{2.51}$$

so on the basis of (2.51) we have

$$\int_t^{t+dt} d\tau \left\{a(\tau) - K(\tau)c(\tau)\right\} e(\tau) = \left\{a(t) - K(t)c(t)\right\} e(t)dt + O\left(dt^2\right). \qquad (2.52)$$

It should be stressed that the functions $a(\tau)$, $K(\tau)$, $c(\tau)$, $e(\tau)$ are assumed to be at least once differentiable. The functions $\xi(\tau)$ and $\eta(\tau)$ may not be differentiable because they behave as white noise, so we may not use the procedure (2.51) for them. From (2.48), (2.49) and (2.52) we have

$$e(t+dt) = \left[1 + \left\{a(t) - K(t)c(t)\right\} dt\right] e(t) + \int_t^{t+dt} d\tau \xi(\tau) - \\ - \int_t^{t+dt} d\tau K(\tau)\eta(\tau) + O\left(dt^2\right). \qquad (2.53)$$

It should be stressed that the first three terms in (2.53) are independent ones because the only random processes on the right-hand side of (2.53) are $e(t)$, $\xi(\tau)$ and $\eta(\tau)$, these processes are independent, see (2.45). After squaring and averaging the cross terms "die" (see the detailed description in Sec. 3.6) and we obtain

$$\left\langle e^2(t+dt)\right\rangle = \left[1 + \left\{a(t) - K(t)c(t)\right\} dt\right]^2 \left\langle e^2(t)\right\rangle + \left\langle \left(\int_t^{t+dt} d\tau \xi(\tau) \right)^2 \right\rangle + \\ + \left\langle \left(\int_t^{t+dt} d\tau K(\tau)\eta(\tau) \right)^2 \right\rangle + O\left(dt^2\right). \qquad (2.54)$$

On the basis of (1.13), (2.43) and (2.51) we obtain

$$\left\langle \left(\int_t^{t+dt} d\tau \xi(\tau) \right)^2 \right\rangle = \int_t^{t+dt} d\tau \int_t^{t+dt} d\tau' \left\langle \xi(\tau)\xi(\tau') \right\rangle = \int_t^{t+dt} d\tau \int_t^{t+dt} d\tau' R_\xi(\tau,\tau') = \\ = \int_t^{t+dt} d\tau \int_t^{t+dt} d\tau' Q(\tau)\delta(\tau - \tau') = \int_t^{t+dt} d\tau Q(\tau) = Q(t)dt + O\left(dt^2\right). \qquad (2.55)$$

Similarly, we obtain

$$\left\langle \left(\int_t^{t+dt} d\tau K(\tau)\eta(\tau) \right)^2 \right\rangle = K^2(t)R(t)dt + O\left(dt^2\right), \qquad (2.56)$$

it is assumed that $Q(t)$ and $R(t)$ are at least once differentiable.

So on the basis of (2.54) – (2.56) we obtain

$$\left\langle e^2(t+dt)\right\rangle = f\left[K(t)\right] = \left[1+2\left\{a(t)-K(t)c(t)\right\}dt\right]\left\langle e^2(t)\right\rangle + Q(t)dt + \\ +K^2(t)R(t)dt + O\left(dt^2\right). \tag{2.57}$$

The aim of the filter is to minimize $\left\langle e^2(t+dt)\right\rangle$, i.e.

$$f\left[K(t)\right] \to \min. \tag{2.58}$$

We will use an idea similar to that used for the minimization of the functional (2.25). Let $K(t)$ be the function that minimizes the functional $f\lfloor K(t)\rfloor$. If we take another function in the form $K(t)+\varepsilon g(t)$ where ε is an arbitrary number and $g(t)$ is an arbitrary function, then the following inequality must hold:

$$\forall\varepsilon,\ \forall g(t)\quad f\lfloor K(t)\rfloor \le f\lfloor K(t)+\varepsilon g(t)\rfloor, \tag{2.59}$$

because the function $K(t)$ minimizes the functional $f\lfloor K(t)\rfloor$. It should be stressed that we deal only with the zeroth and the first orders in dt, the term $O\left(dt^2\right)$ is negligibly small. Neglecting $O\left(dt^2\right)$, we have

$$\forall\varepsilon,\ \forall g(t)\quad f\lfloor K(t)+\varepsilon g(t)\rfloor - f\lfloor K(t)\rfloor = \\ = -2\varepsilon g(t)dt\left[c(t)\left\langle e^2(t)\right\rangle - K(t)R(t)\right] + \varepsilon^2 g^2(t)R(t)dt \ge 0. \tag{2.60}$$

From (2.43) and (1.13) we have

$$R_\eta(t,t) = \left\langle \eta(t)\eta(t)\right\rangle = \left\langle \eta^2(t)\right\rangle = R(t)\delta(0), \tag{2.61}$$

obviously $\left\langle \eta^2(t)\right\rangle \ge 0$, $\delta(0) = +\infty > 0$, so $R(t)$ is non-negative, which leads to the fact that the term on the right-hand side of (2.60) that contains ε^2 is non-negative. So the term that contains ε must also be non-negative $\forall\varepsilon$ and $\forall g(t)$, which leads to the requirement

$$c(t)\left\langle e^2(t)\right\rangle - K(t)R(t) = 0, \tag{2.62}$$

see the same idea in the description of (2.34). From (2.62) we obtain

$$K(t) = \frac{c(t)\left\langle e^2(t)\right\rangle}{R(t)}, \tag{2.63}$$

so the differential equation for $x^{\text{opt}}(t)$ takes the form

$$\frac{dx^{\text{opt}}(t)}{dt} = \frac{c(t)\left\langle e^2(t)\right\rangle}{R(t)}\left\{z(t) - c(t)x^{\text{opt}}(t)\right\} + a(t)x^{\text{opt}}(t) + b(t)u(t). \tag{2.64}$$

We also need a differential equation for $\left\langle e^2(t)\right\rangle$. On the basis of (2.57) and (2.63) we have

$$\left\langle e^2(t+\Delta t)\right\rangle = \left\langle e^2(t)\right\rangle + + \left[2a(t)\left\langle e^2(t)\right\rangle - \frac{c^2(t)\left\langle e^2(t)\right\rangle^2}{R(t)} + Q(t)\right]\Delta t + O\left(\Delta t^2\right) \tag{2.65}$$

.

which leads to

$$\frac{d\left\langle e^2(t)\right\rangle}{dt} = \lim_{\Delta t\to 0}\frac{\left\langle e^2(t+\Delta t)\right\rangle - \left\langle e^2(t)\right\rangle}{\Delta t} = 2a(t)\left\langle e^2(t)\right\rangle - \\ -\frac{c^2(t)\left\langle e^2(t)\right\rangle^2}{R(t)} + Q(t). \tag{2.66}$$

So the differential equation for $\left\langle e^2(t)\right\rangle$ is

$$\frac{d\left\langle e^2(t)\right\rangle}{dt} = -\frac{c^2(t)}{R(t)}\left\langle e^2(t)\right\rangle^2 + 2a(t)\left\langle e^2(t)\right\rangle + Q(t). \tag{2.67}$$

This is a Riccati differential equation. It should be stressed that in some cases this equation admits an analytical solution.

To summarize the above-mentioned, let us describe the whole algorithm. First of all, the function $\left\langle e^2(t)\right\rangle$ is found on the basis of the Riccati equation (2.67). Then the

filter output $x^{\text{opt}}(t)$ is found on the basis of the differential equation (2.64), which contains the function $\langle e^2(t)\rangle$.

2.4. Transfer function, frequency transfer functions and amplitude-frequency response of a linear stationary filter

Let us have a linear stationary filter. As is known, the filter input $x(t)$ and output $y(t)$ are related as

$$y(t)=\int_{-\infty}^{+\infty} d\xi h(\xi)x(t-\xi) \tag{2.68}$$

where $h(\xi)$ is the filter weight function, see (2.1). All the processes are assumed to be stationary and ergodic. On the basis of (2.68) we have

$$y(t+\tau)=\int_{-\infty}^{+\infty} d\xi h(\xi)x(t+\tau-\xi), \tag{2.69}$$

which with account for (2.68) and (1.23) leads to the following expression for the correlation function of the filter output:

$$R_y(\tau)=\langle y(t+\tau)y(t)\rangle=\int_{-\infty}^{+\infty} d\xi \int_{-\infty}^{+\infty} d\eta h(\xi)h(\eta)\langle x(t+\tau-\xi)x(t-\eta)\rangle. \tag{2.70}$$

From (1.14) we can obtain

$$\begin{aligned}\langle x(t+\tau-\xi)x(t-\eta)\rangle &= \int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} dx_1 dx_2 \omega_2(x_1,t+\tau-\xi;x_2,t-\eta)= \\ =\int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} dx_1 dx_2 \omega_2(x_1,t+\tau-\xi+\eta;x_2,t) &= \langle x(t+\tau-\xi+\eta)x(t)\rangle= \\ &= R_x(\tau-\xi+\eta),\end{aligned} \tag{2.71}$$

so from (2.70) and (2.71) we obtain the relation between the correlation functions of the filter output and input

$$R_y(\tau)=\int_{-\infty}^{+\infty}d\xi\int_{-\infty}^{+\infty}d\eta h(\xi)h(\eta)R_x(\tau-\xi+\eta). \tag{2.72}$$

On the basis of (2.72) and (1.43) we can obtain the relation between the spectral densities of the input and the output signal. Obviously,

$$S_y(\omega)=\int_{-\infty}^{+\infty}d\tau R_y(\tau)e^{-i\omega\tau}=\int_{-\infty}^{+\infty}d\tau\int_{-\infty}^{+\infty}d\eta\int_{-\infty}^{+\infty}d\xi h(\eta)h(\xi)e^{-i\omega\tau}R_x(\tau+\eta-\xi). \tag{2.73}$$

After multiplying the integrand in (2.73) by $e^{i\omega(\xi-\eta)}e^{i\omega(\eta-\xi)}$ and making the following change of variables:

$$\alpha=\eta,\quad \beta=\xi,\quad \gamma=\tau+\eta-\xi \tag{2.74}$$

we can obtain that

$$S_y(\omega)=\int_{-\infty}^{+\infty}d\alpha h(\alpha)e^{i\omega\alpha}\int_{-\infty}^{+\infty}d\beta h(\beta)e^{-i\omega\beta}\int_{-\infty}^{+\infty}d\gamma e^{-i\omega\gamma}R_x(\gamma). \tag{2.75}$$

Obviously,

$$\int_{-\infty}^{+\infty}d\gamma e^{-i\omega\gamma}R_x(\gamma)=S_x(\omega), \tag{2.76}$$

and we should introduce some definitions in order to understand the other multipliers on the right-hand side of (2.75).

By definition, the **transfer function** $W(s)$ of a linear stationary filter is the Laplace transform of its weight function:

$$W(\omega)=\int_{0}^{+\infty}dth(t)e^{-\omega t}. \tag{2.77}$$

By definition, the **frequency transfer function** of a linear stationary filter is the Fourier transform of its weight function. The frequency transfer function is equal to $W(i\omega)$ because

$$W(i\omega)=\int_0^{+\infty} dth(t)e^{-i\omega t}=\int_{-\infty}^{+\infty} dth(t)e^{-i\omega t}, \tag{2.78}$$

here the property (2.6) is taken into account. So

$$\int_{-\infty}^{+\infty} d\beta h(\beta)e^{-i\omega\beta}=W(i\omega),\ \ W^*(i\omega)=\int_{-\infty}^{+\infty} d\alpha h(\alpha)e^{i\omega\alpha}, \tag{2.79}$$

the superscript $*$ denotes the complex conjugation. On the basis of (2.79), (2.76) and (2.75) we see that

$$S_y(\omega)=\left|W(i\omega)\right|^2 S_x(\omega). \tag{2.80}$$

The obtained expression is the well-known relation between the spectral densities of the input and output signals of a linear stationary filter.

The physical meaning of $\left|W(i\omega)\right|$ can be illustrated on the basis of the following example. Let the input and output signals of a linear stationary filter be

$$x(t)=A_{\text{in}}\sin(\omega_1 t),\ \ y(t)=A_{\text{out}}\sin(\omega_1 t+\varphi). \tag{2.81}$$

It should be stressed that the *linear stationary filter may change the amplitude and the phase of a signal, but not its frequency*! Here we will not prove this fundamental fact, but in what follows we will illustrate it by several examples (see, for example, the description at the end of Sec. 2.5.2).

The spectral densities of the signals (2.81) can be written on the basis of (1.61):

$$\begin{aligned} S_x(\omega)&=\frac{A_{\text{in}}^2\pi}{2}\left(\delta(\omega+\omega_1)+\delta(\omega-\omega_1)\right),\\ S_y(\omega)&=\frac{A_{\text{out}}^2\pi}{2}\left(\delta(\omega+\omega_1)+\delta(\omega-\omega_1)\right). \end{aligned} \tag{2.82}$$

So from (2.80) and (2.82) we see that

$$K(\omega)=\left|W(i\omega)\right|=\frac{A_{\text{out}}}{A_{\text{in}}}. \tag{2.83}$$

As can be seen, $\left|W(i\omega)\right|$ is the output-to-input amplitude ratio at the frequency ω. This quantity is called the **amplitude-frequency response** of a linear stationary filter.

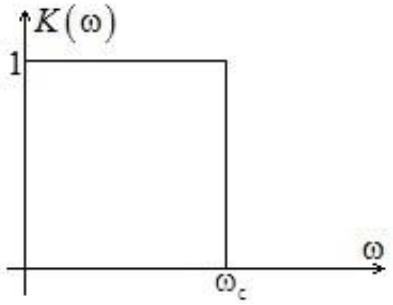
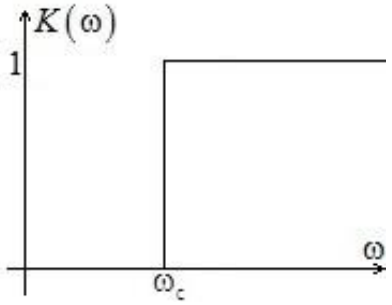
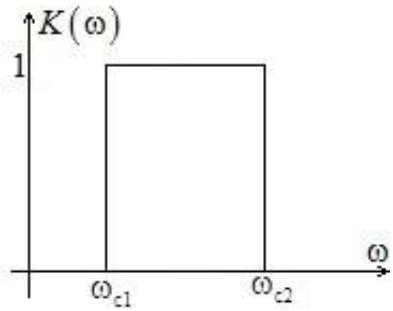
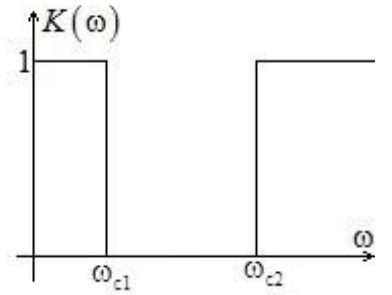
2.5. Linear stationary analog electronic filters

2.5.1. Different types of analog filters

A **low-pass filter** is a filter that passes signals with a frequency lower than a given *cutoff frequency* and attenuates signals with frequencies higher than the cutoff frequency.

A **high-pass filter** is a filter that passes signals with a frequency higher than a given *cutoff frequency* and attenuates signals with frequencies lower than the cutoff frequency.

Table 2.1 – Amplitude-frequency responses for ideal filters

Low-pass filter  ω_c is the cutoff frequency	High-pass filter  ω_c is the cutoff frequency
Band-pass filter  ω_{c1}, ω_{c2} are the cutoff frequencies	Band-stop filter  ω_{c1}, ω_{c2} are the cutoff frequencies

A **band-pass filter** is a filter that passes frequencies within a certain range and attenuates frequencies outside that range.

A **band-stop filter** is a filter that attenuates frequencies within a certain range and passes frequencies outside that range.

The amplitude-frequency response $K(\omega)$ for the corresponding ideal filters are shown in Table 2.1.

But, unfortunately, the ideal filters cannot be realized in practice. In practice the most popular filters are the Butterworth filter, the Chebyshev filters (of type I and type II) and the elliptic filter. ***For simplicity, in what follows we will restrict ourselves only to the description of low-pass filters***.

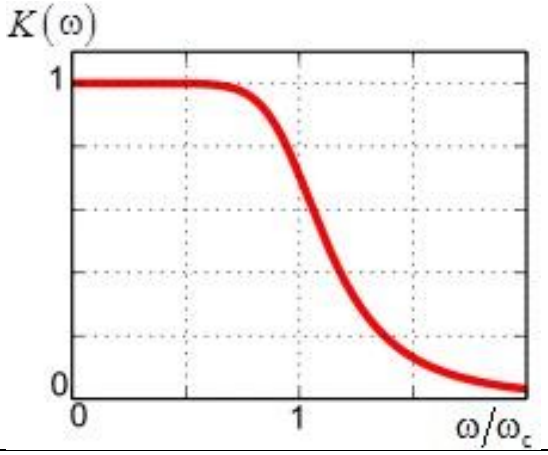
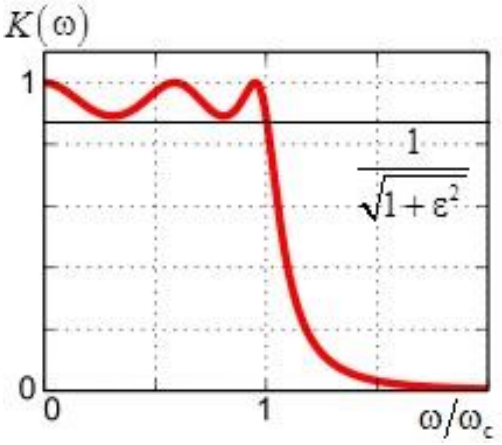
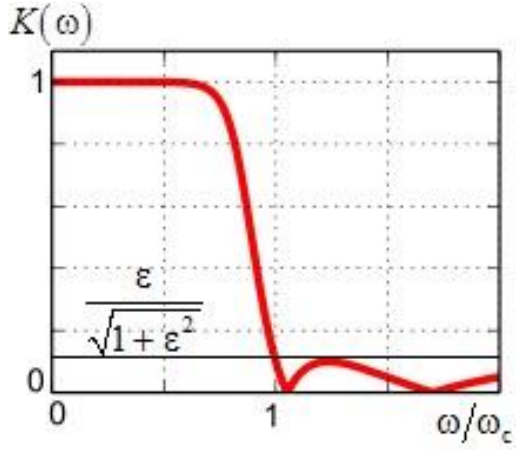
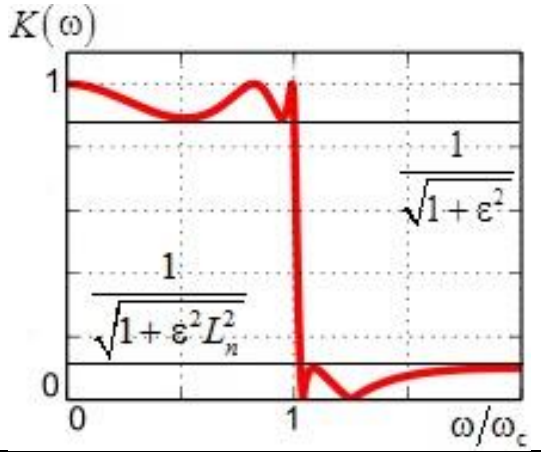
The **Butterworth filter** is designed to have the amplitude-frequency response as flat as possible in the passband. Its amplitude-frequency response contains no ripple in the passband, and in the most of the passband the behavior of the Butterworth filter is rather close to the behavior of the ideal one. In the stopband, there is no ripple either. But the rolloff of the amplitude-frequency response of the Butterworth filter is rather slow, so in the vicinity of the cutoff frequency the behavior of the Butterworth filter is rather far from the ideal one.

The **elliptic filter** has a fast rolloff of the amplitude-frequency response, so in the vicinity of the cutoff frequency the behavior of the elliptic filter may be rather close to the ideal one. But the amplitude-frequency response of the elliptic filter has a ripple both in the passband an in the stopband.

The **Chebyshev filter** is an "intermediate" filter between the Butterworth and the elliptic ones. The rolloff of the Chebyshev filter is faster than that of the Butterworth filter, but slower that of the elliptic filter. The amplitude-frequency response of the Chebyshev filter of type I contains a ripple in the passband, but there is no ripple in the stopband. The amplitude-frequency response of the Chebyshev filter of type II contains a ripple in the stopband, but there is no ripple in the passband.

Schematic graphs and analytical expressions of the amplitude-frequency responses for the corresponding filters are given in Table 2.2.

Table 2.2 – Amplitude-frequency responses for the Butterworth, Chebyshev and elliptic filters. The figures are given for the filters of the same orders.

Butterworth filter of order n $K_n^{\mathrm{B}}(\omega)=\left(1+\left(\omega/\omega_{\mathrm{c}}\right)^{2n}\right)^{-1/2}$ 	Chebyshev filter (type I) of order n $K_n^{\mathrm{CI}}(\omega)=\left[1+\left(\varepsilon\cdot T_n\left(\omega/\omega_{\mathrm{c}}\right)\right)^2\right]^{-1/2}$ 
Chebyshev filter (type II) of order n $K_n^{\mathrm{CII}}(\omega)=\left[1+\left(\varepsilon\cdot T_n\left(\omega_{\mathrm{c}}/\omega\right)\right)^{-2}\right]^{-1/2}$ 	Elliptic filter of order n $K_{\mathrm{n}}^{\mathrm{E}}(\omega)=\left[1+\left(\varepsilon\cdot R_n\left(\xi,\omega/\omega_{\mathrm{c}}\right)\right)^2\right]^{-1/2}$ 

In Table 2.2 the following designations are used: ω_{c} is the cutoff frequency, $\varepsilon\in(0,1)$ is the ripple factor, $T_n(x)$ is the Chebyshev polynomial of the first kind, $\xi>1$ is the selectivity factor, $R_n(\xi,x)$ is the rational elliptic function and $L_n=R_n(\xi,\xi)$.

Let us explain why

$$K_n^{\mathrm{CI}}(\omega)\Big|_{\omega\le\omega_c}\in\left[\frac{1}{\sqrt{1+\varepsilon^2}},1\right],\quad K_n^{\mathrm{CII}}(\omega)\Big|_{\omega\ge\omega_c}\in\left[0,\frac{\varepsilon}{\sqrt{1+\varepsilon^2}}\right]. \tag{2.84}$$

As is known, the explicit form of the Chebyshev polynomials of the first kind is

$$T_n(x)=\frac{1}{2}\left(\left(x+\sqrt{x^2-1}\right)^n+\left(x-\sqrt{x^2-1}\right)^n\right). \tag{2.85}$$

Let us write the first few polynomials on the basis of (2.85):

$$T_0(x)=1,\ T_1(x)=x,\ T_2(x)=2x^2-1,\ \text{etc.} \tag{2.86}$$

Let us show that

$$T_n(\cos\theta)=\cos(n\theta). \tag{2.87}$$

This can be seen as follows:

$$\begin{aligned}T_n(\cos\theta)&=\frac{1}{2}\left(\left(\cos\theta+i\sqrt{1-\cos^2\theta}\right)^n+\left(\cos\theta-i\sqrt{1-\cos^2\theta}\right)^n\right)=\\&=\frac{1}{2}\left(\cos(n\theta)+i\sin(n\theta)+\cos(n\theta)-i\sin(n\theta)\right)=\cos(n\theta).\end{aligned} \tag{2.88}$$

On the basis of (2.87) it can be seen that $|T_n(x)|\le 1$ if $x\in[0,1]$. So

$$T_n(\omega/\omega_c)\Big|_{\omega\le\omega_c}\in[-1,1] \tag{2.89}$$

and

$$K_n^{\mathrm{CI}}(\omega)=\frac{1}{\sqrt{1+\varepsilon^2T_n^2(\omega/\omega_c)}}\in\left[\frac{1}{\sqrt{1+\varepsilon^2}},1\right]\ \text{if}\ \omega\le\omega_c. \tag{2.90}$$

Similarly,

$$K_n^{\mathrm{CII}}(\omega)=\frac{1}{\sqrt{1+\varepsilon^{-2}\left(T_n(\omega_c/\omega)\right)^{-2}}}\in\left[0,\frac{\varepsilon}{\sqrt{1+\varepsilon^2}}\right]\ \text{if}\ \omega\le\omega_c. \tag{2.91}$$

The detailed mathematical description of $K_n^{\mathrm{E}}(\omega)$ is very cumbersome, so here we restrict ourselves only to the description of the most important facts concerning the elliptic filter. The explicit expressions for the first two rational elliptic functions are

$$R_1(\xi,x)=x,\ R_2(\xi,x)=\frac{\left(1+\sqrt{1-\xi^{-2}}\right)x^2-1}{\left(-1+\sqrt{1-\xi^{-2}}\right)x^2+1}. \tag{2.92}$$

The explicit expression for $R_n(\xi,x)$ is very cumbersome, and so it is not given here. Nevertheless, it should be stressed that the following property holds:

$$\lim_{\xi\to\infty} R_n(\xi,x)=T_n(x). \tag{2.93}$$

For example, by a straightforward calculation it can be seen from (2.92) that

$$\lim_{\xi\to\infty} R_2(\xi,x)=2x^2-1=T_2(x), \tag{2.94}$$

see also (2.86). On the basis of (2.93) and the explicit expressions given in Table 2.2 one can see that if $\xi\to\infty$, then the elliptic filter coincides with the Chebyshev filter of type I. If ξ is finite, then the rolloff of the elliptic filter is faster than the rolloff of the Chebyshev filter with the same ripple factor. The peak-to-peak ripple in the passband of the elliptic filter depends only on the ripple factor, while the corresponding peak-to-peak ripple in the stopband depends both on the ripple and on the selectivity factors. The closer is the selectivity factor to unity, the faster is the elliptic filter rolloff.

2.5.2. Electronic Butterworth filter of order 2

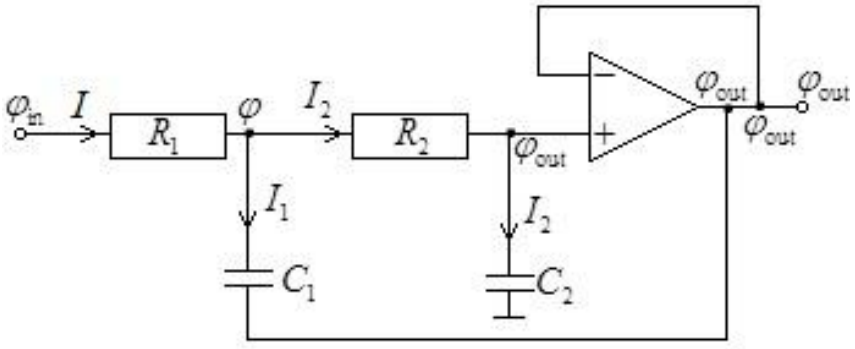


Figure 2.1

In this subsection we describe an electronic realization of the Butterworth filter of order 2. A schematic of this filter is given in Fig. 2.1; φ_{in} is the input potential, φ_{out} is the output potential. The operational amplifier is considered to be

ideal. It equalizes its "+" and "–" input potentials, and its input current is equal to zero.

Obviously, the circuit in Fig 2.1 is described by the following set of equations:

$$\begin{gathered} I(t)=I_1(t)+I_2(t),\ \varphi_{\text{in}}(t)-\varphi(t)=I(t)\cdot R_1,\ \varphi(t)-\varphi_{\text{out}}(t)=q_1(t)/C_1, \\ \dot{q}_1(t)=I_1(t),\ \varphi(t)-\varphi_{\text{out}}(t)=I_2(t)\cdot R_2,\ \varphi_{\text{out}}(t)-0=\frac{q_2(t)}{C_2},\ \dot{q}_2(t)=I_2(t), \end{gathered} \tag{2.95}$$

$q_{1,2}$ are the charges of the capacitors $C_{1,2}$, respectively; here and in what follows the designations

$$\dot{f}(t)=df(t)/dt,\ \ddot{f}(t)=d^2f(t)/dt^2 \tag{2.96}$$

are used. After expressing all the time-dependent quantities in (2.95) in terms of $q_2(t)$, the following differential equation for $q_2(t)$ and the following expression of $\varphi_{\text{out}}(t)$ in terms of $q_2(t)$ can be obtained:

$$\ddot{q}_2(t)C_1R_2R_1+\dot{q}_2(t)(R_1+R_2)+\frac{q_2(t)}{C_2}=\varphi_{\text{in}}(t),\ \varphi_{\text{out}}(t)=\frac{q_2(t)}{C_2}. \tag{2.97}$$

Let the input potential be

$$\varphi_{\text{in}}(t)=A\cos(\omega t). \tag{2.98}$$

Then eq. (2.97) can be rewritten as

$$\begin{gathered} \ddot{q}_2(t)+a\dot{q}_2(t)+bq_2(t)=c\cos(\omega t), \\ a=\frac{(R_1+R_2)}{C_1R_2R_1}>0,\ b=\frac{1}{R_2R_1C_1C_2}>0,\ c=\frac{A}{C_1R_2R_1}. \end{gathered} \tag{2.99}$$

This is a non-uniform second-order differential equation with constant coefficients.

Let us recall how one can solve equations of this type. As is known, the general solution of the equation

$$a\ddot{x}(t)+b\dot{x}(t)+cx(t)=0,\ a,b,c=\text{const} \tag{2.100}$$

is

$$x(t)=\begin{cases} A_1 e^{k_1 t} + A_2 e^{k_2 t}, b^2 > 4ac \\ A_1 t e^{k_1 t} + A_2 e^{k_1 t}, b^2 = 4ac \\ e^{\eta t}\left(A_1 \cos(\xi t) + A_2 \sin(\xi t)\right), b^2 < 4ac \end{cases} \quad (2.101)$$

where A_1, A_2 are arbitrary constants, $k_{1,2} \in \mathbb{R}$ are the roots of the characteristic equation

$$ak^2 + bk + c = 0 \quad (2.102)$$

in the case where $b^2 \geq 4ac$, and $\eta \pm i\xi$ ($\eta, \xi \in \mathbb{R}$) are the roots of Eq. (2.102) in the case where $b^2 < 4ac$. If we have a non-uniform equation

$$a\ddot{x}(t) + b\dot{x}(t) + cx(t) = f(t),\ a,b,c = \text{const}, \quad (2.103)$$

then the general solution of (2.103) is the sum of the general solution of the corresponding uniform equation (2.100) and a partial solution of (2.103).

So let us consider the uniform equation which corresponds to (2.99):

$$\ddot{q}_2(t) + a\dot{q}_2(t) + bq_2(t) = 0 \quad (2.104)$$

The corresponding characteristic equation is

$$k^2 + ak + b = 0, \quad (2.105)$$

and in the case where $a^2 \geq 4b$ the roots are as follows:

$$k_{1,2} = \frac{-a \pm \sqrt{a^2 - 4b}}{2} < 0, \quad (2.106)$$

this inequality takes place because $a > 0$ and $b > 0$. So both roots are negative, and according to (2.101) in the case where $a^2 \geq 4b$ the general solution of (2.104) attenuates exponentially. In the case where $a^2 < 4b$ the roots of (2.105) are

$$k_{1,2} = -\frac{a}{2} \pm \frac{i}{2}\sqrt{4b - a^2},\ \operatorname{Re} k_{1,2} < 0, \quad (2.107)$$

so according to (2.101) in this case the general solution of (2.104) also attenuates exponentially. *To summarize the above-mentioned, the general solution of the uniform differential equation which corresponds to Eq. (2.99) attenuates*

exponentially in any case. This means that that the solution is negligible after the relaxation time.

Let us seek a partial solution of (2.99) in the form

$$q_2(t) = \alpha\cos(\omega t) + \beta\sin(\omega t). \tag{2.108}$$

After taking the first and second derivatives of (2.108) and substituting the results into (2.99) one can obtain

$$\cos(\omega t)\left\{\alpha\left(b-\omega^2\right)+\beta\cdot a\omega\right\}+\sin(\omega t)\left\{\beta\left(b-\omega^2\right)-\alpha\cdot a\omega\right\}=c\cos(\omega t), \tag{2.109}$$

which leads to the following set of linear algebraic equations

$$\alpha\left(b-\omega^2\right)+\beta\cdot a\omega=c,\ \ \beta\left(b-\omega^2\right)-\alpha\cdot a\omega=0. \tag{2.110}$$

The solution of (2.110) is

$$\alpha = c\frac{b-\omega^2}{a^2\omega^2+\left(b-\omega^2\right)^2},\ \ \beta = c\frac{a\omega}{a^2\omega^2+\left(b-\omega^2\right)^2}, \tag{2.111}$$

so the partial solution of (2.99) is (2.108) with the coefficients (2.111). As can be seen, this solution is not-attenuating, so this solution is the solution of (2.99) ***in a steady state***, i.e. after the relaxation time of the solution of (2.104). In what follows only a steady state of the system is considered.

Expression (2.108) with account for (2.111) can be rewritten as

$$q_2(t) = \frac{c}{\gamma}\left(\frac{b-\omega^2}{\gamma}\cos(\omega t)+\frac{a\omega}{\gamma}\sin(\omega t)\right),\ \ \gamma=\sqrt{a^2\omega^2+\left(b-\omega^2\right)^2}. \tag{2.112}$$

The following equality holds:

$$\left(\frac{b-\omega^2}{\gamma}\right)^2+\left(\frac{a\omega}{\gamma}\right)^2=1, \tag{2.113}$$

so we can introduce a parameter φ as follows:

$$\frac{b-\omega^2}{\gamma}=\cos\varphi,\ \frac{a\omega}{\gamma}=\sin\varphi,\ \tan\varphi=\frac{a\omega}{b-\omega^2} \tag{2.114}$$

and rewrite (2.112) as

$$q_2(t)=\frac{c}{\sqrt{a^2\omega^2+\left(b-\omega^2\right)^2}}\cos\left(\omega t-\varphi\right). \tag{2.115}$$

Then according to (2.97) we obtain the output signal

$$\varphi_{\text{out}}(t)=\frac{c}{C_2\sqrt{a^2\omega^2+\left(b-\omega^2\right)^2}}\cos\left(\omega t-\varphi\right), \tag{2.116}$$

and from here it is obvious that the amplitude-frequency response of the filter is

$$K(\omega)=\frac{c}{AC_2\sqrt{a^2\omega^2+\left(b-\omega^2\right)^2}}. \tag{2.117}$$

On the basis of (2.99) this result can be rewritten as

$$\begin{aligned}K(\omega)&=\frac{1}{\sqrt{C_2^2\left(R_1+R_2\right)^2\omega^2+\left(1-C_1C_2R_1R_2\omega^2\right)^2}}=\\&=\left(1+\left[C_2^2\left(R_1+R_2\right)^2-2C_1C_2R_1R_2\right]\omega^2+\left(C_1C_2R_1R_2\right)^2\omega^4\right)^{-1/2}.\end{aligned} \tag{2.118}$$

As can be seen from Table 2.2, the amplitude-frequency response of the Butterworth filter of order 2 is

$$K(\omega)=\left(1+\left(\omega/\omega_c\right)^4\right)^{-1/2}, \tag{2.119}$$

so by comparing (2.118) and (2.119) one can obtain that

$$C_2^2\left(R_1+R_2\right)^2=2C_1C_2R_1R_2,\ \ \omega_c=1/\sqrt{C_1C_2R_1R_2}. \tag{2.120}$$

To summarize the above-mentioned, *if the first equality in (2.120) holds, then the circuit in Fig. 2.1 is an electronic analog Butterworth filter of order 2 with the cutoff frequency* $\omega_c=\left(C_1C_2R_1R_2\right)^{-1/2}$.

Now on the basis of an example of the Butterworth electronic filter we will explain why the linear stationary filter does not change the frequency of a signal. The filter is described by the linear differential equation with the input potential on the right-hand side (see (2.97)), and the steady-state solution of such an equation is of the

frequency which coincides with the frequency of the input potential. As can be seen from the following sections, the same physical picture takes place in other linear stationary filters. So a linear stationary filter cannot change the signal frequency.

2.5.3. Electronic Chebyshev filter (type I) of order 2

In this subsection we describe an electronic realization of the Chebyshev filter (type I) of order 2. Let us consider the circuit in Fig. 2.2.

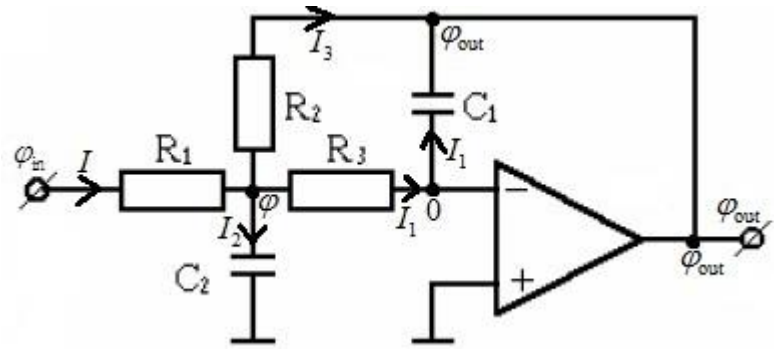


Figure 2.2

In this circuit φ_{in} is the input potential, φ_{out} is the output potential, and the operational amplifier is considered to be ideal. The circuit is described by the following set of equations:

$$\begin{gathered} I(t)=I_1(t)+I_2(t)+I_3(t),\ \ \dot{q}_1(t)=I_1(t),\ \ \dot{q}_2(t)=I_2(t), \\ \varphi_{\text{in}}(t)-\varphi(t)=I(t)\cdot R_1,\ \ \varphi(t)-0=\frac{q_2(t)}{C_2},\ \ 0-\varphi_{\text{out}}(t)=\frac{q_1(t)}{C_1}, \\ \varphi(t)-0=I_1(t)R_3,\ \varphi(t)-\varphi_{\text{out}}(t)=I_3(t)R_2, \end{gathered} \tag{2.121}$$

$q_{1,2}$ are the charges of the capacitors $C_{1,2}$, respectively. By expressing all the time-dependent quantities in (2.121) in terms of $q_1(t)$ and after putting

$$\varphi_{\text{in}}(t)=A\cos(\omega t) \tag{2.122}$$

one can obtain the following:

$$\begin{gathered} \ddot{q}_1(t)+a\dot{q}_1(t)+bq_1(t)=c\cos(\omega t),\ \varphi_{\text{out}}(t)=-q_1(t)/C_1, \\ a=\frac{1}{C_2}\left(\frac{1}{R_1}+\frac{1}{R_2}+\frac{1}{R_3}\right),\ \ b=\frac{1}{C_1C_2R_3R_2},\ c=\frac{A}{C_2R_3R_1}. \end{gathered} \tag{2.123}$$

As can be seen from the previous section, the steady-state solution of eq. (2.123) is

$$q_1(t)=\frac{c}{\sqrt{a^2\omega^2+\left(b-\omega^2\right)^2}}\cos\left(\omega t-\varphi\right),\ \tan\varphi=\frac{a\omega}{b-\omega^2}. \tag{2.124}$$

On the basis of (2.123) and (2.124) it is obvious that

$$\varphi_{\text{out}}(t)=\frac{c}{C_1\sqrt{a^2\omega^2+\left(b-\omega^2\right)^2}}\cos\left(\omega t-\varphi+\pi\right), \tag{2.125}$$

and

$$K(\omega)=\frac{c}{AC_1\sqrt{a^2\omega^2+\left(b-\omega^2\right)^2}}. \tag{2.126}$$

After substituting the explicit expressions for a, b and c from (2.123) into (2.126) by a lengthy straightforward calculation one can obtain that

$$K(\omega)=\left[x^4+\left\{\left(R_1+R_3+\frac{R_3R_1}{R_2}\right)^2\frac{C_1}{C_2R_3R_1}-2\frac{R_1}{R_2}\right\}x^2+\left(\frac{R_1}{R_2}\right)^2\right]^{-1/2}, \tag{2.127}$$

$$x=\omega\sqrt{C_1C_2R_3R_1}.$$

As can be seen from Table 2.2 and (2.86), the explicit expression for the amplitude-frequency response of a Chebyshev filter (type I) of order 2 is

$$K(\omega)=\left[1+\varepsilon^2\left(2\frac{\omega^2}{\omega_{\text{c}}^2}-1\right)^2\right]^{-1/2}=\left[4\varepsilon^2\left(\frac{\omega}{\omega_{\text{c}}}\right)^4-4\varepsilon^2\left(\frac{\omega}{\omega_{\text{c}}}\right)^2+1+\varepsilon^2\right]^{-1/2}. \tag{2.128}$$

The expressions (2.127) and (2.128) should coincide. For example, such coincidence can be obtained as follows. Let us assume that $\omega_{\text{c}}=\left(C_1C_2R_3R_1\right)^{-1/2}$, then $x=\omega/\omega_{\text{c}}$ and

$$K(\omega)=\left[4\varepsilon^2x^4-4\varepsilon^2x^2+1+\varepsilon^2\right]^{-1/2},\ x=\omega/\omega_{\text{c}}. \tag{2.129}$$

Then by equating the coefficients multiplying x^4, x^2 and x^0 in (2.129) and (2.127) we can obtain the following expressions:

$$4\varepsilon^2 = 1,\ 4\varepsilon^2 = 2\frac{R_1}{R_2} - \left(R_1 + R_3 + \frac{R_3 R_1}{R_2}\right)^2 \frac{C_1}{C_2 R_3 R_1},\ \left(\frac{R_1}{R_2}\right)^2 = \varepsilon^2 + 1, \qquad (2.130)$$

which lead to

$$\varepsilon = \frac{1}{2},\ \sqrt{5} - 1 = \left(R_1 + R_3 + \frac{\sqrt{5}}{2} R_3\right)^2 \frac{C_1}{C_2 R_3 R_1},\ \frac{R_1}{R_2} = \frac{\sqrt{5}}{2}. \qquad (2.131)$$

To summarize the above-mentioned, *if the equalities (2.131) hold, then the circuit in Fig. 2.2 is a Chebyshev filter (type I) of order 2 with the cutoff frequency* $\omega_c = \left(C_1 C_2 R_3 R_1\right)^{-1/2}$ *and ripple factor* $\varepsilon = 0.5$.

For example, if we put $R_1 = R_3 = R$ and $C_2 = C$ in Fig 2.2, then the other circuit parameters should be as follows:

$$R_2 = 2R/\sqrt{5},\ \ C_1 = 4\left(\sqrt{5} - 1\right)C \Big/ \left(4 + \sqrt{5}\right)^2. \qquad (2.132)$$

It should also be stressed that a Butterworth filter of order 2 can be designed on the basis of the circuit in Fig. 2.2.

2.5.4. Electronic Chebyshev filter (type II) of order 2

Let us consider the circuit in Fig. 2.3.

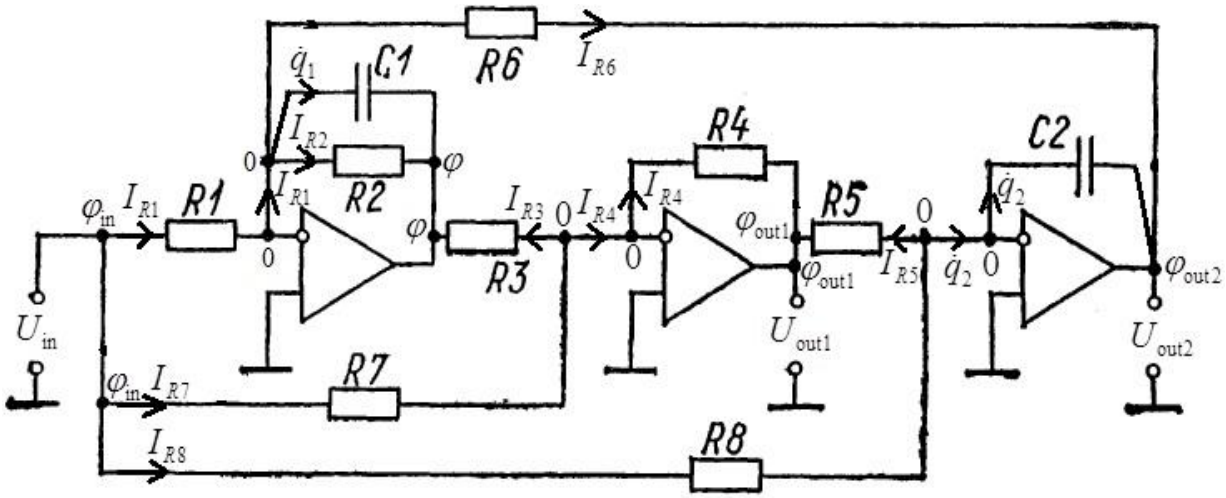


Figure 2.3.

It should be stressed that if $R_7 = R_8 = \infty$. then a Chebyshev filter of type I or a Butterworth filter can be designed on the basis of this circuit, in such a case the output potential is φ_{out2}. If $R_7 R_2 = R_1 R_3$, then a Chebyshev filter of type II or an elliptic filter can be designed on the basis of this circuit, and in such a case the output potential is φ_{out1}. This subsection is devoted to the construction of a Chebyshev filter (type II) of order 2. The circuit in Fig. 2.3 can be described by the following equations:

$$\begin{gathered}
\varphi_{\text{in}}(t) - 0 = I_{R1}(t) R_1 = I_{R7}(t) R_7 = I_{R8}(t) R_8, \\
0 - \varphi(t) = I_{R2}(t) R_2 = I_{R3}(t) R_3 = q_1(t)/C_1, \\
0 - \varphi_{\text{out1}}(t) = I_{R4}(t) R_4 = I_{R5}(t) R_5,\ 0 - \varphi_{\text{вых2}}(t) = I_{R6}(t) R_6 = q_2(t)/C_2, \\
I_{R1}(t) = I_{R2}(t) + I_{R6}(t) + \dot{q}_1(t), \\
I_{R7}(t) = I_{R3}(t) + I_{R4}(t),\ I_{R8}(t) = I_{R5}(t) + \dot{q}_2(t)
\end{gathered} \tag{2.133}$$

where I_{Rj} is the current through the resistor R_j, $j = \overline{1,8}$, and $q_{1,2}$ are the charges of the capacitors $C_{1,2}$, respectively. After expressing all the time-dependent quantities in (2.133) in terms of $q_2(t)$ by a cumbersome calculation one can obtain the following differential equation for $q_2(t)$:

$$\begin{gathered}
\ddot{q}_2(t) + \dot{q}_2(t)\frac{1}{C_1 R_2} + q_2(t)\frac{R_4}{C_1 C_2 R_3 R_5 R_6} = \\
= \varphi_{\text{in}}(t)\frac{R_4}{C_1 R_3 R_5}\left(\frac{1}{R_1} - \frac{R_3}{R_2}\frac{1}{R_7} + \frac{R_3 R_5}{R_2 R_8 R_4}\right) - \dot{\varphi}_{\text{in}}(t)\frac{R_4}{R_5}\left(\frac{1}{R_7} - \frac{R_5}{R_8 R_4}\right).
\end{gathered} \tag{2.134}$$

As indicated above, in the case of construction of a Chebyshev filter of type II or an elliptic filter, the following equality should be valid:

$$R_7 R_2 = R_1 R_3. \tag{2.135}$$

On the basis of (2.135) by putting

$$\varphi_{\text{in}}(t) = A\cos(\omega t) \tag{2.136}$$

one can obtain that (2.134) can be rewritten as

$$\ddot{q}_2(t)+a\dot{q}_2(t)+bq_2(t)=c\cos(\omega t)+d\sin(\omega t),$$
$$a=\frac{1}{C_1R_2}>0,\ b=\frac{R_4}{C_1C_2R_3R_5R_6}>0,\ c=\frac{A}{C_1R_2R_8},\ d=A\omega\frac{R_4}{R_5}\left(\frac{1}{R_7}-\frac{R_5}{R_8R_4}\right). \tag{2.137}$$

The solution of the corresponding uniform equation exponentially attenuates; see the corresponding description in Sec. 2.5.2. So in a steady state the general solution of (2.137) coincides with the partial solution of (2.137) which is sought in the form

$$q_2(t)=\alpha\cos(\omega t)+\beta\sin(\omega t). \tag{2.138}$$

After taking the first and second derivatives of (2.138) and substituting the results into (2.137) one can obtain

$$\cos(\omega t)\left\{\alpha\left(b-\omega^2\right)+\beta\cdot a\omega\right\}+\sin(\omega t)\left\{\beta\left(b-\omega^2\right)-\alpha\cdot a\omega\right\}= \\ =c\cos(\omega t)+d\sin(\omega t), \tag{2.139}$$

so we have the following set of linear algebraic equations

$$\alpha\left(b-\omega^2\right)+\beta\cdot a\omega=c,\ \ \beta\left(b-\omega^2\right)-\alpha\cdot a\omega=d, \tag{2.140}$$

whose solution is

$$\alpha=\frac{c\left(b-\omega^2\right)-da\omega}{a^2\omega^2+\left(b-\omega^2\right)^2},\ \ \beta=\frac{ca\omega+d\left(b-\omega^2\right)}{a^2\omega^2+\left(b-\omega^2\right)^2}. \tag{2.141}$$

The output potential $\varphi_{\text{out1}}(t)$ can be expressed in terms of $q_2(t)$ on the basis of (2.133):

$$\varphi_{\text{out1}}(t)=-R_5\left(\frac{\varphi_{\text{in}}(t)}{R_8}-\dot{q}_2(t)\right). \tag{2.142}$$

On the basis of (2.138), (2.141), (2.142) and (2.136) the following result for the output potential may be obtained:

$$\varphi_{\text{out1}}(t)=R_5\sqrt{\left(\beta\omega-\frac{A}{R_8}\right)^2+\alpha^2\omega^2}\cdot\cos(\omega t+\varphi),\ \tan\varphi=\frac{\alpha\omega}{\beta\omega-\left(A/R_8\right)}, \tag{2.143}$$

the derivation of (2.143) is similar to that of (2.115). Expression (2.143) leads to the fact that the amplitude-frequency response of the system is

$$K(\omega)=\frac{R_5}{A}\sqrt{\left(\beta\omega-\frac{A}{R_8}\right)^2+\alpha^2\omega^2}\,. \tag{2.144}$$

After substituting the expressions for α, β, a, b, c, d from (2.141) and (2.137) into (2.144) by a cumbersome calculation one can obtain that

$$K(\omega)=\left|\frac{R_4}{R_7}\omega^2-\frac{R_4}{C_1C_2R_3R_6R_8}\right|\Bigg/\sqrt{\frac{1}{R_2^2C_1^2}\omega^2+\left(\frac{R_4}{C_1C_2R_3R_5R_6}-\omega^2\right)^2}\,. \tag{2.145}$$

The obtained expression is the explicit dependence of $K(\omega)$ for the circuit under consideration in the case where $R_7R_2=R_1R_3$.

As can be seen from Table 2.2 and from (2.86), the explicit expression for the amplitude-frequency response of the Chebyshev filter (type II) of order 2 is

$$\begin{aligned}K(\omega)&=\left[1+\frac{1}{\varepsilon^2\left(2\left(\omega_c^2/\omega^2\right)-1\right)^2}\right]^{-1/2}=\\&=\frac{\varepsilon\left|2\omega_c^2-\omega^2\right|}{\sqrt{4\varepsilon^2\omega_c^4-4\omega_c^2\varepsilon^2\cdot\omega^2+\left(\varepsilon^2+1\right)\omega^4}}.\end{aligned} \tag{2.146}$$

As can be seen from (2.146) and (2.145), an *ideal Chebyshev filter of type II cannot be built on the basis of the circuit under consideration*, because the coefficient multiplying ω^4 under the square root in the denominator of (2.146) equals ε^2+1 while the same coefficient in (2.145) is equal to 1. *Nevertheless, on the basis of this circuit we can obtain a system whose characteristics are very close to the characteristics of a Chebyshev filter of type II.* To see this, let us put

$$\varepsilon=\frac{R_4}{R_7},\quad \omega_c=\sqrt{\frac{R_7}{2C_1C_2R_6R_3R_8}} \tag{2.147}$$

and

$$R_8 = R_5,\ \frac{1}{C_1 R_2^2} = \frac{2R_4}{C_2 R_6 R_3 R_8}\left(1 - \frac{R_4}{R_7}\right),\ \frac{R_4}{R_7} \ll 1. \tag{2.148}$$

In such a case expression (2.145) can be rewritten as

$$K(\omega) = \frac{\varepsilon\left|\omega^2 - 2\omega_c^2\right|}{\sqrt{4\varepsilon^2\omega_c^4 - 4\varepsilon^2\omega_c^2\omega^2 + \omega^4}}, \tag{2.149}$$

and the only difference between (2.149) and (2.146) is the coefficient multiplying ω^4 within the square root in the denominator. If the circuit parameters are such that $\varepsilon = R_4/R_7 \ll 1$, then $\varepsilon^2 + 1 \approx 1$ and the expressions (2.149) and (2.146) are approximately equal.

So, to summarize the above-mentioned, *if (2.148) and (2.135) hold, then with a good accuracy the circuit in Fig. 2.3 is a Chebyshev filter of type II with the ripple factor* ε *and the cutoff frequency* ω_c *described by expressions (2.147). The output potential of the circuit is* φ_{out1}.

For example, if the parameters R_{1-6} and C_2 are given and the inequality

$$\frac{R_1 R_3}{R_2} \gg R_4 \tag{2.150}$$

holds, then all the other circuit parameters should be taken as

$$R_7 = \frac{R_1 R_3}{R_2},\ R_8 = R_5,\ C_1 = \left[\frac{2R_4 R_2^2}{C_2 R_6 R_3 R_8}\left(1 - \frac{R_4}{R_7}\right)\right]^{-1} \tag{2.151}$$

in order to construct the above-mentioned Chebyshev filter of type II.

2.5.5. Electronic elliptic filter of order 2

As mentioned in the previous section, the circuit in Fig. 2.3 with the output potential φ_{out1} can be a basis for the construction of an elliptic filter if equality (2.135) holds. This subsection is based on the corresponding result (2.145) for the amplitude-frequency response of the circuit under consideration.

As can be seen from Table 2.2 and (2.92), the explicit expression for the amplitude-frequency response of an elliptic filter of order 2 is

$$K(\omega)=\left[1+\varepsilon^2\left(\frac{(1+\mu)\omega^2-\omega_c^2}{(-1+\mu)\omega^2+\omega_c^2}\right)^2\right]^{-1/2}=\left|-(1-\mu)\omega^2+\omega_c^2\right|\times$$
$$\times\left[\left\{(-1+\mu)^2+\varepsilon^2(1+\mu)^2\right\}\omega^4+2\omega_c^2\left\{(-1+\mu)-\varepsilon^2(1+\mu)\right\}\omega^2+\right.$$
$$\left.+\omega_c^4\left(1+\varepsilon^2\right)\right]^{-1/2},\ \mu=\sqrt{1-\xi^{-2}}\,. \tag{2.152}$$

The result (2.145) can be rewritten as

$$K(\omega)=\left|-\frac{R_4}{R_7}\omega^2+\frac{R_4}{C_1C_2R_6R_3R_8}\right|\times\left[\omega^4+\left(\frac{1}{C_1^2R_2^2}-2\frac{R_4}{C_1C_2R_6R_3R_5}\right)\omega^2+\right.$$
$$\left.+\left(\frac{R_4}{C_1C_2R_6R_3R_5}\right)^2\right]^{-1/2}. \tag{2.153}$$

By equating (2.152) and (2.153) we obtain

$$\frac{R_4}{R_7}=1-\mu,\ \omega_c^2=\frac{R_4}{C_1C_2R_6R_3R_8},\ (-1+\mu)^2+\varepsilon^2(1+\mu)^2=1,$$
$$2\omega_c^2\left\{(-1+\mu)-\varepsilon^2(1+\mu)\right\}=\frac{1}{C_1^2R_2^2}-2\frac{R_4}{C_1C_2R_6R_3R_5}, \tag{2.154}$$
$$\omega_c^4\left(1+\varepsilon^2\right)=\left(\frac{R_4}{C_1C_2R_6R_3R_5}\right)^2.$$

From (2.154) and (2.153) we can derive that

$$\left(\frac{R_8}{R_5}\right)^2=1+\frac{1-(R_4/R_7)^2}{(2-(R_4/R_7))^2},\ \frac{1}{C_1R_2^2}=\frac{2R_4}{C_2R_6R_3R_8}\left[\frac{R_8}{R_5}-\frac{R_4}{R_7}-\frac{1-(R_4/R_7)^2}{2-(R_4/R_7)}\right], \tag{2.155}$$

and

$$\frac{R_8}{R_5}=\sqrt{1+\varepsilon^2},\ \frac{R_4}{R_7}=1-\sqrt{1-\xi^{-2}},\ \omega_c=\sqrt{\frac{R_4}{C_1C_2R_6R_3R_8}}\,. \tag{2.156}$$

To summarize the above-mentioned, *if equalities (2.155) and (2.135) hold, then the circuit in Fig. 2.3 is an elliptic filter of order 2 with the ripple factor* ε, *selectivity factor* ξ *and the cutoff frequency* ω_c *given by the expressions (2.156)*.

For example, if the parameters $R_{1-3,5,6}$, ξ and C_2 are given, then all the other circuit parameters should be taken as

$$R_7 = \frac{R_1 R_3}{R_2},\ R_4 = R_7\left(1-\sqrt{1-\xi^{-2}}\right),\ R_8 = R_5\sqrt{1+\frac{1-\left(R_4/R_7\right)^2}{\left(2-\left(R_4/R_7\right)\right)^2}},$$
$$C_1 = \left\{\frac{2R_2^2 R_4}{C_2 R_6 R_3 R_8}\left[\frac{R_8}{R_5}-\frac{R_4}{R_7}-\frac{1-\left(R_4/R_7\right)^2}{2-\left(R_4/R_7\right)}\right]\right\}^{-1} \tag{2.157}$$

in order to construct the above-mentioned elliptic filter.

Chapter 3. Digital Adaptive Filters

3.1 Algorithm of linear digital filtering

The linear stationary system converts the input signal $x(t)$ into the output $y(t)$, equal to the convolution of the function $x(t)$ and the impulse response $h(t)$.

The impulse response of the system $h(t)$ is the response of the system to the input signal $\delta(t)$.

The convolution of two functions $x(t)$, $h(t)$ is as follows:

$$y(t) = \int_{-\infty}^{+\infty} x(t)h(t-\tau)d\tau \qquad (3.1)$$

A linear digital filter (DF) is a discrete system (program or physical device) that converts the x_k sequence of numeric samples of the input signal into a sequence y_k of output signal samples.

The impulse response of the DF is a discrete signal h_k, which is the response of the DC to the "single impulse" (1,0,0,...):

$$(1,0,0,...) \Rightarrow (h_0, h_1, h_2, ...).$$

DF is a linear one, if the sum of the input signals, multiplied by arbitrary coefficients, is converted into a sum of responses to individual terms:

$$\alpha_1 x_k^{(1)} + \alpha_2 x_k^{(1)} + ... + \alpha_n x_k^{(N)} \Rightarrow \alpha_1 y_k^{(1)} + \alpha_2 y_k^{(1)} + ... + \alpha_n y_k^{(N)}$$

for any coefficients α_1, α_2, …, α_N.

A linear DF is stationary if, when an input single pulse is shifted by any number of sampling intervals, the impulse response shifts in the same way, without changing in form:

$$(0,1,0,0,...) \Rightarrow (0, h_0, h_1, h_2, ...),\ (0,0,1,0,0,...) \Rightarrow (0,0, h_0, h_1, h_2, ...),\ \ldots$$

The properties of linearity and stationarity imply a general linear digital filtering algorithm: let $x_k = (x_0, x_1, x_2, ...)$ be some signal at the input of the DF with a known impulse response, then, based on the properties of linearity and stationarity, the m-th sample of the output signal y_k:

$$y_m = \sum_{k=0}^{m} x_k h_{m-k} \,. \qquad (3.2)$$

Expression (3.2) has the following meaning: at the time of each sample, the DF performs a weighted summation of all previous values of the input signal, and the "weights" are counts of the impulse response. That is, the DF has some "memory" in relation to past input influences. For physically realizable DFs, the impulse response cannot be non-zero at the points preceding the input pulse impulse time.

3.2 Frequency Coefficient of Transmission. System function. Impulse response

Frequency Coefficient of Transmission of DF

Let the harmonic sequence of the form: $x_k = Ae^{i(\omega k\Delta+\varphi)}$, unlimitedly extended in time (k = 0, ±1, ±2, ..) be fed to the input of a linear DF. Define the output signal of the DF:

$$y_m = \sum_{k=-\infty}^{\infty} x_k h_{m-k} = \sum_{k=-\infty}^{\infty} Ae^{i(\omega k\Delta+\varphi)} h_{m-k} = Ae^{i\varphi} \sum_{k=-\infty}^{\infty} e^{i\omega k\Delta} h_{m-k} = \{n = m-k\} =$$

$$= Ae^{i\varphi} \sum_{n=-\infty}^{\infty} e^{i\omega(m-n)\Delta} h_n = Ae^{i\varphi} e^{i\omega m\Delta} \sum_{n=-\infty}^{\infty} e^{-i\omega n\Delta} h_n = x_m \sum_{n=-\infty}^{\infty} e^{-i\omega n\Delta} h_n \,,$$

which with account for the fact that $h_{n<0} = 0$ leads to

$$y_m = x_m \sum_{n=0}^{\infty} e^{-i\omega n\Delta} h_n \,.$$

That is, the output samples are obtained from the input by multiplying by the complex value $K(i\omega)$:

$$K(i\omega) = \sum_{n=0}^{\infty} e^{-i\omega n\Delta} h_n \,. \qquad (3.3)$$

where $K(i\omega)$ is the **frequency coefficient of transmission** of the DF.

Analyzing (3.3) we get: $K(i\omega)$ is a periodic function of frequency with a period equal to the sampling frequency $\omega_d = 2\pi/\Delta$; $K(i\omega)$ depends on the impulse response of the system.

System function of DF

Let us associate with discrete signals x_k, y_k, h_k their Z-transformations $X(z)$, $Y(z)$, $H(z)$, respectively. The output signal y_k is a convolution of the input signal x_k and the impulse response h_k, then, based on the 3rd property of the Z-transform, the function $Y(z) = H(z)X(z)$ corresponds to the output signal. The system function $H(z)$ of a stationary linear DF is the ratio of Z - conversion of the output signal to Z - transformation of the signal at the input:

$$H(z) = \frac{Y(z)}{X(z)} = \sum_{k=0}^{\infty} h_k z^{-k}. \qquad (3.4)$$

that is, the system function of the DF is the Z-transform of the impulse response.

In order to obtain the frequency coefficient of transmission of the DF from the system function, it is necessary to make a substitution in (3.4): $z = e^{i\omega\Delta}$.

Example

DF has impulse response $h_k = (1, -1, 0, 0, ...)$, find the system function and transmission coefficient of the DF.

$$H(z) = \sum_{k=0}^{\infty} h_k z^{-k} = h_0 + \frac{h_1}{z} + \frac{h_2}{z^2} + ... = 1 + \frac{-1}{z} = 1 - z,\ K(i\omega) = 1 - e^{-i\omega\Delta}.$$

3.3 Filters with finite impulse response

Filters of this type work in accordance with the algorithm:

$$y_j = a_0 x_j + a_1 x_{j-1} + a_2 x_{j-2} ... + a_m x_{j-m} \qquad (3.5)$$

where a_0, a_1, a_2, ... a_m is the sequence of coefficients, m is the order of DF.

A non-recursive FC performs weighted summation of previous input signal samples and does not use past output signal samples.

Apply the Z-transformation to both sides of (3.5), then the **system function of the non-recursive DF** will be:

$$H(z) = \frac{Y(z)}{X(z)} = \frac{a_0 z^m + a_1 z^{m-1} + a_2 z^{m-2} + ... + a_m}{z^m}. \qquad (3.6)$$

The main blocks of the DF are the blocks of delay of the sampling values by one sampling interval Z^{-1}, and the large-scale blocks that perform the multiplication operations by the corresponding coefficients a_i in digital form.

Algorithm of functioning of non-recursive DF:

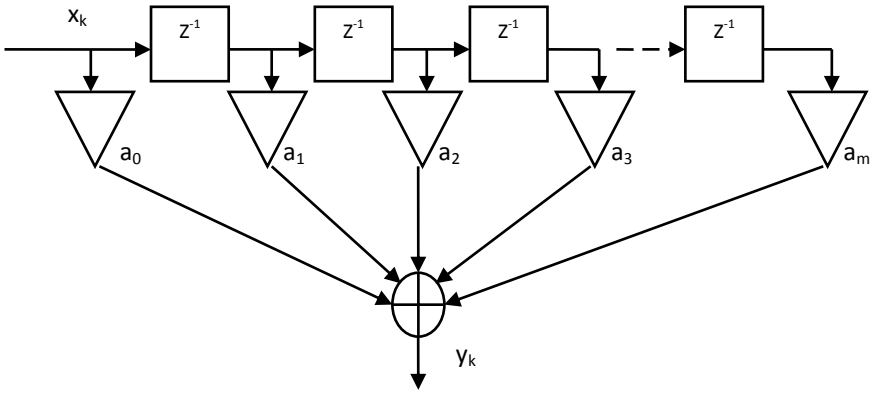


Figure 3.1 - Block diagram of non-recursive DF

The transverse structure of the DF gave the second name to the non-recursive DF - transversal.

From the outputs of the large-scale blocks, the signals go to the adder, where the output is a count of the output signal.

Based on (3.6), the impulse response of the non-recursive DF can be determined. Analysis (3.6) shows that each term of the function $H(z)$ makes a contribution equal to the corresponding coefficient a_n, shifted by n positions in the direction of delay, then the **impulse response of the non-recursive DF** is:

$$h_k = \left(a_0, a_1, ..., a_m\right), \tag{3.7}$$

moreover, the impulse response of the non-recursive DF contains a finite number of terms.

In the expression for the system function, let's introduce a change of variable $z = e^{i\omega\Delta}$, then **the frequency coefficient of transmission of the non-recursive DF** will be:

$$K\left(i\omega\right) = a_0 + a_1 e^{-i\omega\Delta} + a_2 e^{-2i\omega\Delta} + ... + a_m e^{-mi\omega\Delta}. \tag{3.8}$$

3.4 Filters with infinite impulse response

To form the *i*-th sample of the output signal y_i, the previous values of not only the input signal, but also the output signal are used:

$$y_j = a_0 x_j + a_1 x_{j-1} + a_2 x_{j-2} \ldots + a_m x_{j-m} + b_1 y_{j-1} + b_2 y_{j-2} + \ldots + b_n y_{j-n} \qquad (3.9)$$

where the coefficients $b_1, \ldots, b_n$, which determine the recursive part of the filtering algorithm, are not equal to zero simultaneously. Recursion is a mathematical method consisting in cyclically referring to data obtained in the preceding stages.

System function of the recursive DF

Perform a *Z* - transformation over (3.9), then the system function will be:

$$H(z) = \frac{Y(z)}{X(z)} = \frac{a_0 z^n + a_1 z^{n-1} + a_2 z^{n-2} + \ldots + a_m z^{n-m}}{z^n - b_1 z^{n-1} - b_2 z^{n-2} - \ldots - b_n}. \qquad (3.10)$$

The structural scheme of the recursive DF is presented in Fig. 3.2. The upper part of the structural scheme corresponds to the non-recursive part of the filtering algorithm. Its implementation requires, in general, *m*+1 scale blocks (multiplication operations) and *m* memory cells in which the input samples are stored.

The recursive part of the algorithm corresponds to the lower part of the structural scheme, where *n* consecutive values of the output signal are used, which in the process of filter operation move from cell to cell by shifting.

3.5 Criteria for the operation of adaptive filters

The main purpose of the adaptive filter is to find its own parameters, as a rule, the vector of weight coefficients (WC) $h_N(k)$, at which the output signal $y(k)$ minimizes the given target function. The target function is usually a function of the input, output, and desired signals, i.e. $F = f\left[x(k), y(k), d(k)\right]$. It must be non-negative and real, i.e. $F \geq 0$ for all $x(k)$, $y(k)$ and $d(k)$. The adaptive algorithm minimizes the target function so that the output signal of the adaptive filter approximates the desired signal, and $h_N(k) \to h_{N,O}$ where $h_{N,O}$ is coordinate vector of

the minimum of the multidimensional target function in the space of the WC. The target function can be considered as a function of errors, i.e. $F = f\left[e(k)\right]$.

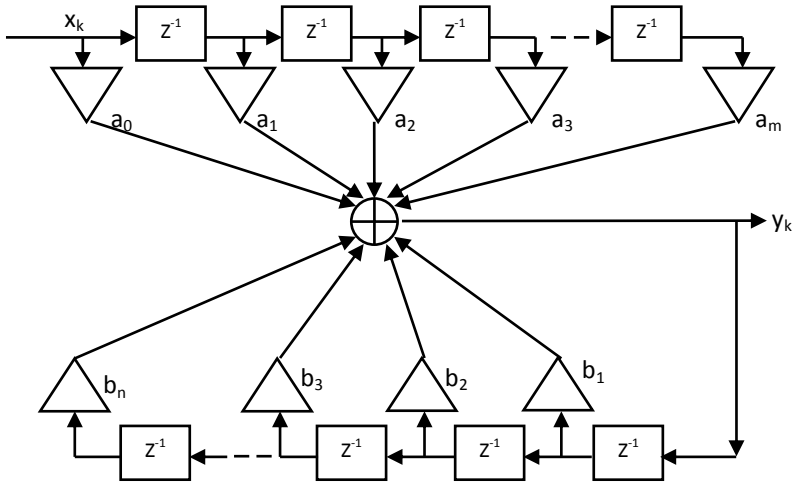


Figure 3.2 - Structural scheme of the recursive DF

There are many different ways to determine the target function of an adaptive filter, which ultimately affect the complexity of its minimization algorithms. The the most commonly used target functions are as follows:

1. $F = f\left[e(k)\right] = \left\langle \left|e\left(k\right)\right|^2 \right\rangle$ – standard error (MSE),

2. $F = f\left[e(k)\right] = \sum_{t=1}^{k} \left|e\left(t\right)\right|^2$ – least squares (Least Squares, LS),

3. $F = f\left[e(k)\right] = \sum_{t=1}^{k} \lambda^{k-t} \left|e\left(t\right)\right|^2$ – weighted least squares (Weighted Least Squares, WLS),

4. $F = f\left[e(k)\right] = \left|e\left(k\right)\right|^2$ – squared instant error.

Achieving a minimum of the targetpower function is a criterion for the operation of the adaptive filter.

Other criteria are also used, for example, the criterion of constancy of the module of information symbols (Constance Modulus, CM) used in adaptive signal processing in digital communication systems. This criterion is also based on minimization of the error function between the modulus value (amplitude) of the

output signal of the adaptive filter and the known modulus value (envelope) of information symbols raised to a certain power. This minimization ensures that the output envelope of the adaptive filter approaches the known constant envelope value of the information symbols. Therefore, this criterion got its name for the desired end result, i.e. the output signal envelope of the adaptive filter, which allows one to correctly recognize (detect) received information symbols in this signal.

The choice of quadratic target functions is largely due to the fact that as a result of their use, adaptive filtering algorithms can be obtained in the form of recurrent calculations that do not contain logical operations, which distinguishes such algorithms from general-purpose computing procedures. In addition, these target[-

„ functions are unimodal, which guarantees the convergence of adaptive filtering algorithms based on them to a single optimal solution.

Strictly speaking, the MSE function is only a convenient mathematical concept, since its computation requires an infinite amount of data, since obtaining this function implies averaging over the ensemble of realizations of the observed signals. MSE-function 1 is used, for example, in optimal Wiener filtering.

The target functions 2 - 4 differ in both the complexity of implementing adaptive algorithms based on them, and the convergence characteristics, and the residual errors in the steady state of these algorithms. Thus, the square of the instantaneous error is the simplest, in terms of the implementation of algorithms, target function. But algorithms based on this function are characterized by slow convergence due to the highly simplified target function. The LS target function is usually used when processing stationary signals, and the WLS one is usually used when processing slowly changing signals. Adaptive algorithms that use target functions 3 and 4 are often referred to as least-squares algorithms.

3.6 Scalar discrete Kalman filter

For clarity, let us consider a simple example, which is not quite physical. Let us have a radio-controlled car whose motion is one-dimensional, i.e. it can move only forward or backwards. It is assumed that the instantaneous velocity of the car is

completely controlled by a joystick. Of course, such an example is not physical because the velocity cannot change instantly. Nevertheless, we consider it because it clearly demonstrates the idea of the scalar Kalman filter.

In the example under consideration, the motion of the car is completely described by one parameter which is its coordinate x. Let the coordinate be measured at time instants $t_n = n \cdot \Delta t$ where Δt is a small time interval between two neighboring instants. In fact, we have a discrete set of the measurements $x_n = x(t_n)$.

Physically, the coordinate varies as follows:

$$x_{n+1} = x_n + \upsilon_n \Delta t \tag{3.11}$$

where $\upsilon_n = \upsilon(t_n)$ is the velocity at the instant t_n that is "governed" by the joystick. Equation (3.11) describes the change in the coordinate in the "ideal" case where the motion of the car is completely controlled by the joystick. But there are some small perturbations acting on the car (wind, road irregularities, and so on). With account for such perturbations, we have

$$x_{n+1} = x_n + \upsilon_n \Delta t + \xi_n \tag{3.12}$$

where ξ_n is the term which is due to the above-mentioned perturbations, ξ_n is a discrete random process.

Let the coordinate be measured by a sensor. The sensor reading z_n is

$$z_n = x_n + \eta_n \tag{3.13}$$

where the random discrete process η_n is the sensor error. For example, it may occur due to radio noise.

In fact, we have a set of the measured values z_n, and we need a filter that generates the true coordinate values x_n. This filter is called the Kalman filter.

The above-mentioned example is given just for clarity. Now let us describe the scalar Kalman filter more generally.

Let us have a system described by one parameter x that changes discretely, and on the basis of some physical considerations the "ideal" law of change in this parameter can be written as

$$x_{t+1} = ax_t + bu_t \tag{3.14}$$

where u_t is the known quantity that governs the system evolution; a, b are some known constants (in the above-mentioned example $u_t = \upsilon_t$, $a = 1$, $b = \Delta t$). The parameter x is measured by a device whose readings are z_t. Let η_t be a random stationary process whose physical meaning is the device error. Let ξ_t be a random stationary process that describes the random character of the system evolution. So we have

$$x_{t+1} = ax_t + bu_t + \xi_t, \quad z_t = cx_t + \eta_t \tag{3.15}$$

where the constant c is the known device gain coefficient.

In fact, the input for the Kalman filter is the set z_t. ***The aim of the filter is to generate the output set*** x_t^{opt} ***closest to the set*** x_t***. The word "closest" should be understood as "having the smallest mean-square deviation".***

Let us denote the filter error as e_t: $\quad e_t = x_t - x_t^{\text{opt}}$. (3.16)

The following assumptions are made:

1. The variances σ_ξ^2 *and* σ_η^2 *of the processes* ξ_t *and* η_t*, respectively, are known.*

2. The average values of the processes ξ_t *and* η_t *are zeros:*

$$\langle \xi_t \rangle = 0, \quad \langle \eta_t \rangle = 0. \tag{3.17}$$

3. The processes ξ_t*,* η_t *and the filter error* e_t *are independent, i.e.*

$$\begin{gathered} \forall t_1, t_2 \quad \langle \xi_{t_1} e_{t_2} \rangle = \langle \xi_{t_1} \rangle \langle e_{t_2} \rangle = 0, \quad \langle \eta_{t_1} e_{t_2} \rangle = \langle \eta_{t_1} \rangle \langle e_{t_2} \rangle = 0, \\ \langle \xi_{t_1} \eta_{t_2} \rangle = \langle \xi_{t_1} \rangle \langle \eta_{t_2} \rangle = 0. \end{gathered} \tag{3.18}$$

It should be stressed that (3.18) follows from (3.17).

The Kalman filter is based on a recurrent algorithm. Let us suppose that we have calculated x_t^{opt} and $\langle e_t^2 \rangle$. We have to calculate x_{t+1}^{opt} and $\langle e_{t+1}^2 \rangle$. We seek x_{t+1}^{opt} in the following form:

$$x_{t+1}^{\text{opt}} = Kz_{t+1} + (1 - Kc)\left(ax_t^{\text{opt}} + bu_t\right) \tag{3.19}$$

where K is an unknown constant called the Kalman coefficient. It should be stressed that the Kalman coefficient may be different for different steps of the algorithm. As mentioned above, the following mean-square deviation should be minimized:

$$\left\langle (x_{t+1} - x_{t+1}^{\text{opt}})^2 \right\rangle = \left\langle e_{t+1}^2 \right\rangle \to \min. \tag{3.20}$$

The Kalman coefficient K should be found on the basis of the condition (3.20).

From (3.19) and (3.15) we have

$$\begin{gathered} e_{t+1} = x_{t+1} - x_{t+1}^{\text{opt}} = ax_t + bu_t + \xi_t - Kz_{t+1} - (1-Kc)\left(ax_t^{\text{opt}} + bu_t\right) = \\ = ax_t + bu_t + \xi_t - K\left(cx_{t+1} + \eta_{t+1}\right) - (1-Kc)\left(ax_t^{\text{opt}} + bu_t\right) = \\ = ax_t + bu_t + \xi_t - K\left(c\left(ax_t + bu_t + \xi_t\right) + \eta_{t+1}\right) - (1-Kc)\left(ax_t^{\text{opt}} + bu_t\right) \end{gathered} \tag{3.21}$$

which after removing brackets with account for (3.16) leads to

$$e_{t+1} = (1-Kc)ae_t + (1-Kc)\xi_t - \eta_{t+1}K. \tag{3.22}$$

From (3.22) we have

$$e_{t+1}^2 = (1-Kc)^2 a^2 e_t^2 + (1-Kc)^2 \xi_t^2 + \eta_{t+1}^2 K^2 + G_t \tag{3.23}$$

where G_t are the so-called "cross terms":

$$G_t = 2a(1-Kc)^2 \xi_t e_t - 2a(1-Kc)Ke_t\eta_{t+1} - 2(1-Kc)K\xi_t\eta_{t+1}. \tag{3.24}$$

It should be stressed that the only random processes in (3.22) – (3.24) are η_t, e_t and ξ_t, so on the basis of (3.18) and (3.23) we have

$$\begin{gathered} \left\langle G_t \right\rangle = 2a(1-Kc)^2 \left\langle \xi_t e_t \right\rangle - 2a(1-Kc)K\left\langle e_t\eta_{t+1} \right\rangle - 2(1-Kc)K\left\langle \xi_t\eta_{t+1} \right\rangle = 0, \\ \left\langle e_{t+1}^2 \right\rangle = (1-Kc)^2 a^2 \left\langle e_t^2 \right\rangle + (1-Kc)^2 \left\langle \xi_t^2 \right\rangle + K^2 \left\langle \eta_{t+1}^2 \right\rangle. \end{gathered} \tag{3.25}$$

From (3.25) it is clear why we do seek x_{t+1}^{opt} in the form (3.19)! This form leads to the expression (3.22) with three independent terms on the right-hand side, and after squaring and averaging the cross terms "die".

Obviously on the basis of the fact that η_t is a stationary process and from (3.17) we have

$$\sigma_\xi^2 = \left\langle \xi_t^2 \right\rangle - \left\langle \xi_t \right\rangle^2 = \left\langle \xi_t^2 \right\rangle, \ \sigma_\eta^2 = \left\langle \eta_t^2 \right\rangle - \left\langle \eta_t \right\rangle^2 = \left\langle \eta_t^2 \right\rangle = \left\langle \eta_{t+1}^2 \right\rangle, \tag{3.26}$$

which with account for (3.25) leads to

$$\left\langle e_{t+1}^2 \right\rangle = \left(1 - Kc\right)^2 a^2 \left\langle e_t^2 \right\rangle + \left(1 - Kc\right)^2 \sigma_\xi^2 + K^2 \sigma_\eta^2 . \tag{3.27}$$

We should minimize $\left\langle e_{t+1}^2 \right\rangle$, so K is found as follows

$$\frac{\partial \left\langle e_{t+1}^2 \right\rangle}{\partial K} = 0 \Rightarrow K = c \frac{a^2 \left\langle e_t^2 \right\rangle + \sigma_\xi^2}{\sigma_\eta^2 + c^2 a^2 \left\langle e_t^2 \right\rangle + c^2 \sigma_\xi^2} . \tag{3.28}$$

So x_{t+1}^{opt} is calculated by (3.19) where the Kalman coefficient K is given by (3.28). We also have to calculate $\left\langle e_{t+1}^2 \right\rangle$. On the basis of (3.27) and (3.28) by a straightforward calculation it can be shown that

$$\left\langle e_{t+1}^2 \right\rangle = \sigma_\eta^2 \frac{a^2 \left\langle e_t^2 \right\rangle + \sigma_\xi^2}{\sigma_\eta^2 + c^2 \sigma_\xi^2 + c^2 a^2 \left\langle e_t^2 \right\rangle} \tag{3.29}$$

Obviously, we need some initial values for the filter output x^{opt} and for the mean square error $\left\langle e^2 \right\rangle$. They are usually chosen as

$$\left\langle e_0^2 \right\rangle = \sigma_\eta^2, \; x_0^{\text{opt}} = z_0 . \tag{3.30}$$

To summarize the above-mentioned, let us repeat the whole recurrent algorithm. First of all, the initial values for the filter output x^{opt} and for the mean square filter error $\left\langle e^2 \right\rangle$ are specified. Usually they are specified by (3.30). On the basis of the initial values $\left\langle e_0^2 \right\rangle$ and x_0^{opt} the values $\left\langle e_1^2 \right\rangle$ and x_1^{opt} are calculated:

$$\begin{gathered} x_1^{\text{opt}} = Kz_1 + \left(1 - Kc\right)\left(a x_0^{\text{opt}} + b u_0\right), \; K = c \frac{a^2 \left\langle e_0^2 \right\rangle + \sigma_\xi^2}{\sigma_\eta^2 + c^2 a^2 \left\langle e_0^2 \right\rangle + c^2 \sigma_\xi^2} ; \\ \left\langle e_1^2 \right\rangle = \sigma_\eta^2 \frac{a^2 \left\langle e_0^2 \right\rangle + \sigma_\xi^2}{\sigma_\eta^2 + c^2 \sigma_\xi^2 + c^2 a^2 \left\langle e_0^2 \right\rangle} . \end{gathered} \tag{3.31}$$

Then $\left\langle e_2^2 \right\rangle$ and x_2^{opt} are calculated on the basis of x_1^{opt} and $\left\langle e_1^2 \right\rangle$, and so on. At step n, $\left\langle e_n^2 \right\rangle$ and x_n^{opt} are calculated on the basis of the above-calculated x_{n-1}^{opt} and $\left\langle e_{n-1}^2 \right\rangle$ by the formulas

$$x_n^{\text{opt}} = K_n z_n + (1 - K_n c)\left(a x_{n-1}^{\text{opt}} + b u_{n-1}\right),\ K_n = c\frac{a^2\left\langle e_{n-1}^2\right\rangle + \sigma_\xi^2}{\sigma_\eta^2 + c^2 a^2\left\langle e_{n-1}^2\right\rangle + c^2\sigma_\xi^2};$$
$$\left\langle e_n^2\right\rangle = \sigma_\eta^2\frac{a^2\left\langle e_{n-1}^2\right\rangle + \sigma_\xi^2}{\sigma_\eta^2 + c^2\sigma_\xi^2 + c^2 a^2\left\langle e_{n-1}^2\right\rangle}, \tag{3.32}$$

here we write the subscript n on K_n in order to stress that the Kalman coefficient may be different at different steps of the algorithm.

3.7 Vector discrete Kalman filter

In the previous section we considered the case where the system was described by one parameter. The aim of the vector Kalman filter is similar to that of the scalar one, but it deals with a system that is described by more than one parameter.

Let us consider a system described by n parameters x_1, x_2, …, x_n, $n > 1$, that change discretely.

The system vector of state is the column that contains all the parameters that describe the system:

$$x_t = \begin{pmatrix} x_{t,1} \\ x_{t,2} \\ \vdots \\ x_{t,n} \end{pmatrix} \tag{3.33}$$

where the subscript t indicates that the corresponding quantity is taken at the instant t.

In what follows, we will deal with the covariance matrix. The **covariance matrix** $\text{cov}(x)$ of a random vector (column) x is the matrix whose components are defined as

$$\text{cov}(x)_{ij} = \text{cov}(x)_{ji} = \left\langle x_i x_j\right\rangle . \tag{3.34}$$

The discrete vector Kalman filter can be described by a simple analogy with the scalar one:

Table 3.1 – Analogy between the vector and the scalar discrete Kalman filters

Scalar filter	Vector filter
The change in the parameters with time: $x_{t+1} = Ax_t + Bu_t + \xi_t$ x is the parameter describing the system; ξ_t is a random process that describes the random character of the system evolution; u_t is the known quantity that governs the system evolution; A, B are known constants.	*The change in the parameters with time:* $x_{t+1} = Ax_t + Bu_t + \xi_t,$ (3.35) x is the system vector of state; the physical meaning of all the other quantities is the same as in the scalar filter, but it should be mentioned that A and B are $n \times n$ matrices and x_t, u_t, ξ_t are column vectors. Equation (3.35) is written in matrix form!
The device readings: $z_t = Cx_t + \eta_t$ z is the device reading, η_t is the device error, and the constant C is the device gain coefficient.	*The device readings:* $z_t = Cx_t + \eta_t,$ (3.36) the physical meaning of all the quantities is the same as in the scalar filter, but it should be mentioned that C is a $n \times n$ matrix, and x_t, z_t, η_t are column vectors. Equation (3.36) is written in matrix form.
The filter error and the aim of the filter: $e_t = x_t - x_t^{\text{opt}}, \quad \forall t \ \left\langle e_t^2 \right\rangle \to \min.$	*The filter error and the aim of the filter* $e_t = x_t - x_t^{\text{opt}}, \quad \forall t \ \sum_{j=1}^{n} \left\langle e_{t,j}^2 \right\rangle \to \min,$ (3.37) e_t is a column vector.
The assumptions of the theory: σ_ξ^2 and σ_η^2 are known; $\left\langle \xi_t \right\rangle = 0, \quad \left\langle \eta_t \right\rangle = 0;$ $\forall t_1, t_2 \quad \left\langle \xi_{t_1} e_{t_2} \right\rangle = \left\langle \xi_{t_1} \right\rangle \left\langle e_{t_2} \right\rangle = 0,$	*The assumptions of the theory:* $\text{cov}(\xi_t)$ and $\text{cov}(\eta_t)$ are known, $\forall i = 1,2,..,n \ \left\langle \xi_{t,i} \right\rangle = 0 \ \left\langle \eta_{t,i} \right\rangle = 0;$ (3.38) $\forall i, j = 1,2,..,n \quad \forall t_1, t_2$ (3.39)

$\langle \eta_{t_1} e_{t_2} \rangle = \langle \eta_{t_1} \rangle \langle e_{t_2} \rangle = 0,$ $\langle \xi_{t_1} \eta_{t_2} \rangle = \langle \xi_{t_1} \rangle \langle \eta_{t_2} \rangle = 0.$	$\langle \xi_{t_1,i} e_{t_2,j} \rangle = \langle \xi_{t_1,i} \rangle \langle e_{t_2,j} \rangle = 0,$ $\langle \eta_{t_1,i} e_{t_2,j} \rangle = \langle \eta_{t_1,i} \rangle \langle e_{t_2,j} \rangle = 0,$ $\langle \xi_{t_1,i} \eta_{t_2,j} \rangle = \langle \xi_{t_1,i} \rangle \langle \eta_{t_2,j} \rangle = 0.$
The recurrent relation for the output $x_{t+1}^{\text{opt}} = Kz_{t+1} + (1-KC)(ax_t^{\text{opt}} + bu_t),$ the Kalman coefficient K is found from the condition $\langle e_{t+1}^2 \rangle \to \min$	*The recurrent relation for the output* $x_{t+1}^{\text{opt}} = Kz_{t+1} + (I-KC)(Ax_t^{\text{opt}} + Bu_t),$ (3.40) I is an identity matrix, the Kalman coefficient K is a $n \times n$ matrix, and it is found from the condition $\sum_{j=1}^{n} \langle e_{t+1,j}^2 \rangle \to \min.$ (3.41) The recurrent relation (3.40) is given in matrix form.

It should be mentioned that in the general case the matrices A, B and C may depend on time, but for simplicity we consider the case where they are constant ones.

Let us consider the quantity $e_{t+1}^T e_{t+1}$; here and in what follows, the superscript T indicates that the corresponding matrix is transposed. We have

$$e_{t+1}^T e_{t+1} = \begin{pmatrix} e_{t+1,1} & e_{t+1,2} & \dots & e_{t+1,n} \end{pmatrix} \begin{pmatrix} e_{t+1,1} \\ e_{t+1,2} \\ \vdots \\ e_{t+1,n} \end{pmatrix} = e_{t+1,1}^2 + e_{t+1,2}^2 + \dots + e_{t+1,n}^2 = \sum_{j=1}^{n} e_{t+1,j}^2, \qquad (3.42)$$

so from (3.42) it can be seen that the condition (3.41) can be rewritten as

$$\langle e_{t+1}^T e_{t+1} \rangle \to \min. \qquad (3.43)$$

Similarly to the derivation of the formula (3.22) on the basis of (3.35)–(3.37) and (3.40), we can obtain

$$e_{t+1} = (A - KCA)e_t + (I - KC)\xi_t - K\eta_{t+1}, \qquad (3.44)$$

during the derivation of (3.44) and in what follows we should remember that the multiplication of matrices is not commutative, i.e. we are not allowed to change the order of the matrices in their product!

On the basis of the well-known properties

$$\left(AB\right)^T = B^T A^T,\ \left(A+B\right)^T = A^T + B^T \tag{3.45}$$

and from (3.44) we have

$$e_{t+1}^T = e_t^T \left(A - KCA\right)^T + \xi_t^T \left(I - KC\right)^T - \eta_{t+1}^T K^T. \tag{3.46}$$

From (3.46) and (3.44) we can obtain

$$\begin{aligned} e_{t+1}^T e_{t+1} = e_t^T \left(A - KCA\right)^T \left(A - KCA\right) e_t + \\ + \xi_t^T \left(I - KC\right)^T \left(I - KC\right) \xi_t + \eta_{t+1}^T K^T K \eta_{t+1} + G_t \end{aligned} \tag{3.47}$$

where G_t contains the "cross terms":

$$\begin{aligned} G_t = e_t^T \left(A - KCA\right)^T \left(I - KC\right) \xi_t - e_t^T \left(A - KCA\right)^T K \eta_{t+1} + \\ + \xi_t^T \left(I - KC\right)^T \left(A - KCA\right) e_t - \xi_t^T \left(I - KC\right)^T K \eta_{t+1} - \\ - \eta_{t+1}^T K^T \left(A - KCA\right) e_t - \eta_{t+1}^T K^T \left(I - KC\right) \xi_t. \end{aligned} \tag{3.48}$$

First of all, it should be stressed that each term in G_t is a number rather than a matrix. Let us consider the first term in (3.48). e_t^T is a $1 \times n$ matrix, $\left(A - KCA\right)^T$ is an $n \times n$ matrix, $\left(I - KC\right)$ is an $n \times n$ matrix, and ξ_t is an $n \times 1$ matrix, so their product has the structure $\left(1 \times n\right) \cdot \left(n \times n\right) \cdot \left(n \times n\right) \cdot \left(n \times 1\right) = 1 \times 1$. Similarly, it can be shown that each term on the right-hand side of (3.48) is a number. Moreover, similarly it can be shown that each term on the right-hand side of (3.47) is a number.

Let us consider the average value of the first term on the right-hand side of (3.48). By the well-known Einstein summation rules we have

$$e_t^T \left(A - KCA\right)^T \left(I - KC\right) \xi_t = e_{t,i}^T \left(A - KCA\right)_{ij}^T \left(I - KC\right)_{jk} \xi_{t,k}, \tag{3.49}$$

here and in what follows, the summation is performed over the repeated matrix subscripts. So on the basis of (3.49) we obtain

$$\left\langle e_t^T \left(A - KCA\right)^T \left(I - KC\right) \xi_t \right\rangle = \left\langle e_{t,i}^T \left(A - KCA\right)_{ij}^T \left(I - KC\right)_{jk} \xi_{t,k} \right\rangle = \tag{3.50}$$

$$= \left(A - KCA\right)_{ij}^{T} \left(I - KC\right)_{jk} \left\langle e_{t,i}^{T} \xi_{t,k} \right\rangle = 0$$

due to (3.39). Similarly it can be shown that the average values of all the terms in G_t are zeros, so on the basis of (3.47) we have

$$\begin{aligned} \left\langle e_{t+1}^{T} e_{t+1} \right\rangle = f\left(K\right) = \left\langle e_t^{T} \left(A - KCA\right)^{T} \left(A - KCA\right) e_t \right\rangle + \\ + \left\langle \xi_t^{T} \left(I - KC\right)^{T} \left(I - KC\right) \xi_t \right\rangle + \left\langle \eta_{t+1}^{T} K^{T} K \eta_{t+1} \right\rangle . \end{aligned} \tag{3.51}$$

It should be stressed that the result (3.51) justifies the fact that x_{t+1}^{opt} is sought in the form (3.40): after averaging all the cross terms "die".

Now we have to minimize the function of matrix $f\left(K\right)$ due to (3.43). We will use an idea similar to that used for the minimization of the functional (2.25). Let K be the matrix that minimizes the function $f\left(K\right)$. If we take another matrix in the form $K + \varepsilon k$ where ε is an arbitrary number and k is an arbitrary $n \times n$ matrix, then the following inequality must hold:

$$\forall \varepsilon ,\ \forall k \quad f\left(K\right) \le f\left(K + \varepsilon k\right), \tag{3.52}$$

because the matrix K minimizes the function (3.51).

On the basis of (3.51) we have

$$\begin{aligned} f\left(K + \varepsilon k\right) - f\left(K\right) = \left\langle e_t^{T} \left(A - \left(K + \varepsilon k\right) CA\right)^{T} \left(A - \left(K + \varepsilon k\right) CA\right) e_t \right\rangle + \\ + \left\langle \xi_t^{T} \left(I - \left(K + \varepsilon k\right) C\right)^{T} \left(I - \left(K + \varepsilon k\right) C\right) \xi_t \right\rangle + \\ + \left\langle \eta_{t+1}^{T} \left(K + \varepsilon k\right)^{T} \left(K + \varepsilon k\right) \eta_{t+1} \right\rangle - \left\langle e_t^{T} \left(A - KCA\right)^{T} \left(A - KCA\right) e_t \right\rangle - \\ - \left\langle \xi_t^{T} \left(I - KC\right)^{T} \left(I - KC\right) \xi_t \right\rangle - \left\langle \eta_{t+1}^{T} K^{T} K \eta_{t+1} \right\rangle \end{aligned} \tag{3.53}$$

which after removing brackets with account for (3.52) leads to

$$\begin{aligned} \forall \varepsilon ,\ \forall k \quad f\left(K + \varepsilon k\right) - f\left(K\right) = \varepsilon \left(- \left\langle e_t^{T} \left(kCA\right)^{T} \left(A - KCA\right) e_t \right\rangle - \right. \\ - \left\langle e_t^{T} \left(A - KCA\right)^{T} kCAe_t \right\rangle - \left\langle \xi_t^{T} \left(I - KC\right)^{T} kC\xi_t \right\rangle - \\ \left. - \left\langle \xi_t^{T} \left(kC\right)^{T} \left(I - KC\right) \xi_t \right\rangle + \left\langle \eta_{t+1}^{T} k^{T} K \eta_{t+1} \right\rangle + \left\langle \eta_{t+1}^{T} K^{T} k \eta_{t+1} \right\rangle \right) + \\ + \varepsilon^2 \left(\left\langle \left(kCAe_t\right)^{T} \left(kCAe_t\right) \right\rangle + \left\langle \left(kC\xi_t\right)^{T} \left(kC\xi_t\right) \right\rangle + \left\langle \left(k\eta_{t+1}\right)^{T} \left(k\eta_{t+1}\right) \right\rangle \right) \ge 0 . \end{aligned} \tag{3.54}$$

First of all, let us consider the terms multiplying ε^2. k, C and A are $n \times n$ matrices, e_t is an $n \times 1$ matrix, so $kCAe_t$ is a column vector ($n \times 1$ matrix):

$$kCAe_t = \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{pmatrix} \Rightarrow \left(kCAe_t\right)^T kCAe_t = \sum_{j=1}^{n} b_j^2 \geq 0 \ . \tag{3.55}$$

Similarly, it can be shown that

$$\left\langle \left(kC\xi_t\right)^T \left(kC\xi_t\right) \right\rangle \geq 0 , \ \left\langle \left(k\eta_{t+1}\right)^T \left(k\eta_{t+1}\right) \right\rangle \geq 0 , \tag{3.56}$$

so the part of (3.54) that contains ε^2 is non-negative. Let us consider the terms multiplying ε in (3.54). Obviously, each of them is a number rather than a matrix. Let us denote

$$a_1\left(k,K\right) = -\left\langle e_t^T \left(kCA\right)^T \left(A - KCA\right)e_t \right\rangle, \ a_2\left(k,K\right) = -\left\langle e_t^T \left(A - KCA\right)^T kCAe_t \right\rangle,$$

$$a_3\left(k,K\right) = -\left\langle \xi_t^T \left(I - KC\right)^T kC\xi_t \right\rangle, \ a_4\left(k,K\right) = -\left\langle \xi_t^T \left(kC\right)^T \left(I - KC\right)\xi_t \right\rangle, \tag{3.57}$$

$$a_5\left(k,K\right) = \left\langle \eta_{t+1}^T k^T K\eta_{t+1} \right\rangle, \ a_6\left(k,K\right) = \left\langle \eta_{t+1}^T K^T k\eta_{t+1} \right\rangle.$$

In fact, on the basis of (3.54)–(3.57) we can conclude that the matrix K that minimizes the function (3.51) must obey the property

$$\forall \varepsilon , \ \forall k \qquad \varepsilon \sum_{\alpha=1}^{6} a_\alpha\left(k,K\right) \geq 0 . \tag{3.58}$$

Obviously,

$$a_1\left(k,K\right) = a_1^T\left(k,K\right), \tag{3.59}$$

because $a_1\left(k,K\right)$ is a number. At the same time, from (3.57) we have

$$a_1^T\left(k,K\right) = -\left\langle e_t^T \left(A - KCA\right)^T kCAe_t \right\rangle = a_2\left(k,K\right). \tag{3.60}$$

So, on the basis of (3.60) and (3.59) we have

$$a_1\left(k,K\right) = a_2\left(k,K\right). \tag{3.61}$$

Similarly, it can be shown that

$$a_3(k,K)=a_4(k,K),\ a_5(k,K)=a_6(k,K). \tag{3.62}$$

From (3.58), (3.57), (3.61) and (3.62) we have that the matrix K that minimizes the function (3.51) must obey the property

$$\forall\varepsilon,\ \forall k \qquad \varepsilon\left(a_1(k,K)+a_3(k,K)+a_5(k,K)\right)\geq 0. \tag{3.63}$$

The number ε can be either positive or negative, so such a requirement is valid only if the sum in parentheses multiplying ε is zero:

$$-\left\langle e_t^T\left(kCA\right)^T\left(A-KCA\right)e_t\right\rangle-\left\langle \xi_t^T\left(I-KC\right)^T kC\xi_t\right\rangle+\left\langle \eta_{t+1}^T k^T K\eta_{t+1}\right\rangle=0, \tag{3.64}$$

here, the explicit expressions (3.57) are used.

Let us consider the expression $\left\langle \eta_{t+1}^T Q\eta_{t+1}\right\rangle$ where Q is a non-random $n\times n$ matrix. By the Einstein summation rules we have

$$\begin{aligned}\left\langle \eta_{t+1}^T Q\eta_{t+1}\right\rangle=\left\langle \eta_{t+1,i}^T Q_{ij}\eta_{t+1,j}\right\rangle=Q_{ij}\left\langle \eta_{t+1,i}^T\eta_{t+1,j}\right\rangle=Q_{ij}\left\langle \eta_{t+1,j}\eta_{t+1,i}\right\rangle=\\ =Q_{ij}\,\mathrm{cov}\left(\eta_{t+1}\right)_{ji}=\mathrm{Tr}\left(Q\cdot\mathrm{cov}\left(\eta_{t+1}\right)\right),\end{aligned} \tag{3.65}$$

Here and in what follows $\mathrm{Tr}A$ denotes the trace of a matrix A. Similarly,

$$\left\langle \xi_t^T Q\xi_t\right\rangle=\mathrm{Tr}\left(Q\cdot\mathrm{cov}\left(\xi_t\right)\right),\ \left\langle e_t^T Qe_t\right\rangle=\mathrm{Tr}\left(Q\cdot\mathrm{cov}\left(e_t\right)\right). \tag{3.66}$$

On the basis of the relations (3.66) and (3.65) the condition (3.64) can be rewritten as follows:

$$\begin{aligned}\forall k \quad -\mathrm{Tr}\left(\left(kCA\right)^T\left(A-KCA\right)\mathrm{cov}\left(e_t\right)\right)-\mathrm{Tr}\left(\left(kC\right)^T\left(I-KC\right)\mathrm{cov}\left(\xi_t\right)\right)+\\ +\mathrm{Tr}\left(k^T K\,\mathrm{cov}\left(\eta_{t+1}\right)\right)=0\ .\end{aligned} \tag{3.67}$$

On the basis of the well-known properties

$$\mathrm{Tr}\left(AB\right)=\mathrm{Tr}\left(BA\right),\ \mathrm{Tr}\left(A+B\right)=\mathrm{Tr}\left(A\right)+\mathrm{Tr}\left(B\right) \tag{3.68}$$

the expression (3.67) can be rewritten as

$$\begin{aligned}\forall k \quad \mathrm{Tr}\Big[\Big(-\left(A-KCA\right)\mathrm{cov}\left(e_t\right)\left(CA\right)^T-\\ -\left(I-KC\right)\mathrm{cov}\left(\xi_t\right)C^T+K\,\mathrm{cov}\left(\eta_{t+1}\right)\Big)k^T\Big]=0\end{aligned} \tag{3.69}$$

which leads to

$$-\left(A-KCA\right)\mathrm{cov}\left(e_t\right)\left(CA\right)^T-\left(I-KC\right)\mathrm{cov}\left(\xi_t\right)C^T+K\,\mathrm{cov}\left(\eta_{t+1}\right)=0\,. \tag{3.70}$$

From (3.70) by a straightforward calculation it can be shown that

$$K=P_tC^T\left(CP_tC^T+\mathrm{cov}\left(\eta_{t+1}\right)\right)^{-1},\ P_t=A\mathrm{cov}\left(e_t\right)A^T+\mathrm{cov}\left(\xi_t\right). \tag{3.71}$$

It is also very important to calculate $\mathrm{cov}\left(e_{t+1}\right)$ because, as can be seen from (3.71) and (3.40), x_{t+2}^{opt} depends on $\mathrm{cov}\left(e_{t+1}\right)$.

On the basis of (3.44) we have

$$e_{t+1,i}=\left(A-KCA\right)_{ij}e_{t,j}+\left(I-KC\right)_{ij}\xi_{t,j}-K_{ij}\eta_{t+1,j}\,. \tag{3.72}$$

From (3.72), (3.34) and (3.39) we have

$$\begin{gathered}\mathrm{cov}\left(e_{t+1}\right)_{il}=\left\langle e_{t+1,i}e_{t+1,l}\right\rangle=\left(A-KCA\right)_{ij}\mathrm{cov}\left(e_t\right)_{jk}\left(A-KCA\right)_{kl}^T+\\+\left(I-KC\right)_{ij}\mathrm{cov}\left(\xi_t\right)_{jk}\left(I-KC\right)_{kl}^T+K_{ij}\,\mathrm{cov}\left(\eta_{t+1}\right)_{jk}K_{kl}^T.\end{gathered} \tag{3.73}$$

which with account for (3.71) leads to

$$\mathrm{cov}\left(e_{t+1}\right)=\left(I-K_{t+1}C\right)P_t. \tag{3.74}$$

We also need initial conditions for $\mathrm{cov}\left(e\right)$ and x^{opt}. They are usually chosen as

$$\mathrm{cov}\left(e_0\right)=\mathrm{cov}\left(\eta_0\right),\ \ x_0^{\mathrm{opt}}=\mathrm{z}_0\,. \tag{3.75}$$

To summarize the above-mentioned, let us repeat the whole recurrent algorithm. First of all, the initial values for the filter output x^{opt} and for the covariance matrix of the filter error $\mathrm{cov}\left(e_0\right)$ are specified. Usually they are specified by the relations (3.75). Then at each step the filter output and the covariance matrix of the filter error are calculated by the recurrent formulas

$$\begin{gathered}x_{t+1}^{\mathrm{opt}}=K_{t+1}z_{t+1}+\left(I-K_{t+1}C\right)\left(Ax_t^{\mathrm{opt}}+Bu_t\right),\\K_{t+1}=P_tC^T\left(CP_tC^T+\mathrm{cov}\left(\eta_{t+1}\right)\right)^{-1},\ P_t=A\mathrm{cov}\left(e_t\right)A^T+\mathrm{cov}\left(\xi_t\right),\\\mathrm{cov}\left(e_{t+1}\right)=\left(I-K_{t+1}C\right)P_t,\end{gathered} \tag{3.76}$$

here, we write the subscript $t+1$ on K_{t+1} in order to stress that the Kalman coefficient may be different at different steps of the algorithm. It should be stressed

that if the matrices A, B and C depend on time, the derivation of the formulas is the same, and the following result can be obtained:

$$x_{t+1}^{\text{opt}} = K_{t+1} z_{t+1} + \left(I - K_{t+1} C_t\right)\left(A_t x_t^{\text{opt}} + B_t u_t\right),$$
$$K_{t+1} = P_t C_t^T \left(C_t P_t C_t^T + \text{cov}\left(\eta_{t+1}\right)\right)^{-1},\ P_t = A_t \,\text{cov}\left(e_t\right) A_t^T + \text{cov}\left(\xi_t\right), \tag{3.77}$$
$$\text{cov}\left(e_{t+1}\right) = \left(I - K_{t+1} C_t\right) P_t .$$

3.8 Adaptive Filter Structures

Like filters with fixed weighting factors (WC) described earlier (Sec. 3.3, 3.4), there are two main structures of adaptive filters. These are filters with finite impulse response (FIR), or transversal, and filters with infinite impulse response (IIR), or recursive. The structure of the single-channel adaptive FIR filter is shown in Fig. 3.3.

The output signal $y(k)$ of the adaptive filter is formed as linear combination of delayed samples of the input signal $x(k-n+1)$, taken with weights $h_n(k-1)$, calculated at previous iterations *(k–1)* with respect to current iterations k, i.e., as

$$y\left(k\right) = \sum_{n=1}^{N} h_n^*\left(k-1\right) x\left(k-n+1\right) = h_N^H\left(k-1\right) x_N\left(k\right), \tag{3.78}$$

where N is the WC number of the filter,

$$h_N^H\left(k-1\right) = \left(h_1\left(k-1\right), h_2\left(k-1\right), ..., h_{N-1}\left(k-1\right), h_N\left(k-1\right)\right)^t$$

is the WC vector and

$$x_N\left(k\right) = \left(x\left(k\right), x\left(k-1\right), ..., x\left(k-N+2\right), x\left(k-N+1\right)\right)^t$$

is the vector of signals in the filter. The filter order is defined as N–1, i.e., by the number of delay lines. Vectors $h_N(k)$ and $x_N(k)$ are complex in the general case.

Hereinafter, the superscript "*t*" will be used to denote the transposition of a vector or matrix, and the superscript H wil be used to denote the Hermitian conjugation operation, that is, transposition and complex conjugation, denoted by the "*" symbol, of vector or matrix elements. The first subscript N in the notation for vectors and matrices will indicate the number of elements in a vector or the number

of elements in a square matrix, defined as $N \times N$. The number of elements in a rectangular matrix will be denoted by the first two subscripts. For example, the A^t_{NM} entry will denote the transposition of the A_{NM} matrix, i.e., the matrix $B_{MN} = A^t_{NM}$. Thus, the subscripts in the matrix notation will indicate the number of elements of the untransposed matrices. In some cases, the subscript N will also be used in the notation of scalar variables, indicating that these variables are functions of N-dimensional vectors to distinguish like scalar variables that are functions of $(N+1)$-, N- or $(N-1)$-dimensional vectors in fast adaptive filtering algorithms.

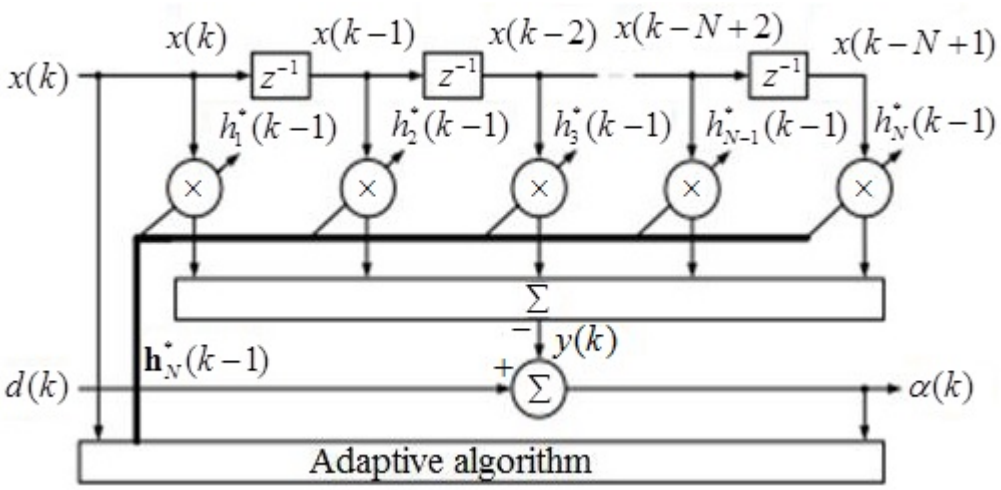


Figure 3.3 - Single-channel adaptive FIR filter

Subscripts in the elements of matrices and vectors will denote the numbers of these elements in matrices and vectors according to the generally accepted numbering system. As other signs in the notation of vectors, matrices or their elements, various symbols in the subscript can be used, followed by a comma after designation of the size of a vector or matrix or the numbers of their elements, as well as symbols in the subscript. The unit matrix (a square diagonal matrix with units on the main diagonal and zero remaining elements) will be denoted by I_N, zero matrices (matrices containing only zero elements) - O_N and O_{NM}, a unit vector (vector containing all

ones) - i_N, and the zero vector (a vector containing only zero elements) is represented by the symbol 0_N. By the vector we will usually understand a column vector.

Using the output signal of the adaptive filter (3.78), it is possible to calculate the signal of a priori simulation error of the required signal $d(k)$ as

$$\alpha(k) = d(k) - y(k) = d(k) - h_N^H(k-1)x_N(k). \qquad (3.79)$$

As you can see, the terms "a priori" and "a posteriori" are associated with the values of the WC of the adaptive filter, calculated respectively at the previous and current iterations of the algorithm of the adaptive filtering.

In practice, during the operation of the adaptive filter, a priori errors (3.79) are observed at its output, since the current value of the output signal of the filter $y(k)$ is generated from the WC values calculated at the previous iteration. A posteriori errors are commonly used in WC calculation algorithms and when forming the target function of an adaptive filter, for example, the mean square error (Mean Square Error, MSE) $F = E\{e(k)e^*(k)\} = E\{|e(k)|^2\}$ where E is the averaging operation over the ensemble of realizations. In the case of FIR filters, such a function is a real unimodal quadratic function in the space of real or complex WC, that is, it is characterized by a single minimum.

We will consider mainly adaptive filters with complex WC, unless otherwise specified separately. This is due to some mathematical subtleties used in obtaining WK calculation algorithms for such filters, and the fact that the transition from adaptive algorithms for filters with real WC to algorithms for filters with complex WC is often not obvious, despite the fact that it reduces mainly to the correct arrangement of operations of complex conjugation of some variables used in algorithms.

Therefore, in this book, the theory of adaptive filtering will be described in relation to filters with complex WC, since in most literature sources adaptive algorithms are considered mainly for filters with real WC. Transition from the description of filters with complex WC to the description of filters with real WC is trivial and usually reduces to the exclusion of all operations of complex conjugation

in matrices, vectors and scalar variables. This transition will be accompanied by the appearance of a fixed factor of 2 in some mathematical expressions, which is absent in the mathematical expressions of algorithms for adaptive filters with complex WC.

Complex signals are processed in adaptive antenna arrays (AAA) or echo-compensators and equalizers of digital quadrature modulated communication systems. This naturally leads to the need to use adaptive filters with complex WC.

In the general case, adaptive FIR filters can be multichannel and, at the same time, contain an unequal number of WCs in the channels (Fig. 3.4). WC vector of such *M*-channel adaptive filter

$$h_N(k-1)=\left(h_{N_1}^t(k-1),h_{N_2}^t(k-1),...,h_{N_{M-1}}^t(k-1),h_{N_M}^t(k-1)\right)^t$$

is formed from a sequence of WC channel vectors

$$h_{N_m}(k)=\left(h_{1,m}(k-1),h_{2,m}(k-1),...,h_{N_{M-1},m}(k-1),h_{N_M,m}(k-1)\right)^t$$

and a vector of signals

$$x_N(k)=\left(x_{N_1}^t(k),x_{N_2}^t(k),...,x_{N_{M-1}}^t(k),x_{N_M}^t(k)\right)^t$$

is formed from a sequence of channel signal vectors

$$x_{N_m}(k)=\left(x_m(k),x_m(k-1),...,x_m(k-N_m+2),x_m(k-N_m+1)\right)^t.$$

The total number of VK multichannel filter is defined as $T=\sum_{m=1}^{M}N_m$.

The unequal number of WCs in the channels of a multichannel adaptive filter, on the one hand, is often dictated by the physical nature of the problem, and on the other hand, due to the always existing limitations on the computational complexity of the implementation of filters. Since in adaptive filters computational complexity is a function of the total number of WC *N*, this number should not be increased, unless there are any reasons, despite the fact that a number of algorithms, for example, for multichannel adaptive filters with the same number of WCs in channels are mathematically (algorithmically) “much simpler” than algorithms for filters with unequal number of WC in channels.

The computational complexity of the DSP algorithms, to which the adaptive filtering algorithms belong, is understood to be the number of arithmetic operations (usually additions, subtractions, multiplications and divisions, and sometimes more

complex operations, such as extracting square roots) required to perform one iteration of the algorithm.

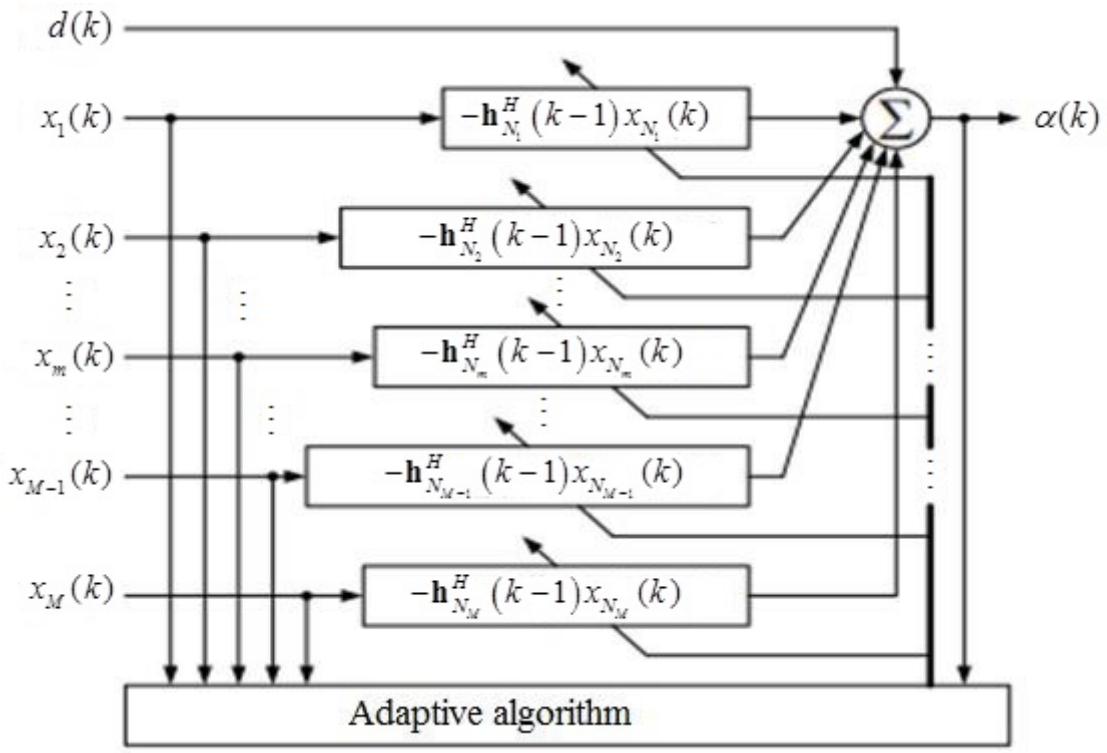


Figure 3.4 - Multichannel adaptive FIR filter

Two particular structures of the adaptive filter should be noted (see Fig. 3.4). This is a multichannel filter with one WC ($N_m = 1$, $m = 1, 2, ..., M$) in each channel, which is used in narrowband AAA, and a multichannel filter with the same number of WC in channels $N_1 = N_2 = \ldots = N_M > 1$, which is used in broadband acoustic gratings or multichannel acoustic echo cancellers. The structure of a multichannel filter of a general form (see Fig. 3.4) is also used when implementing nonlinear polynomial adaptive filters, in which nonlinear cores are sets of multichannel filters with the number of WCs in the channels varying from $N_1 = 1$ to $N_M = N$ with a step equal to one WC. Besides, in addition, a near- and far-echo signal compensator in modems for wired communication channels or an equalizer with feedback can also be considered as a two-channel adaptive filter with a different WC number in the channels.

Adaptive IIR filters to date have not yet found wide application in practice, since, in addition to problems with stability, they are characterized by the problem of multi-extremality (i.e. non-unimodality) of the target function, which does not generally guarantee the convergence of the WC calculation process to the global (i.e. best) solution.

3.9. Gradient methods of search for a function minimum and their use in the filtration theory

Problems of the filtration theory are often reduced to a search for a minimum of a multivariable function. The corresponding numerical search can be realized on the basis of gradient algorithms.

For clarity, let us consider a two-variable function $f(x,y)$. Let us consider x and y as the Cartesian coordinates on a plane. Let us take a point with coordinates (X,Y) and let us move from this point in its small vicinity. As is known, during this movement $f(x,y)$ increases fastest if we move in the direction of the gradient vector, and $f(x,y)$ decreases fastest if we move in the direction opposite to the gradient vector. The gradient vector is as follows:

$$\text{grad}f(x,y)\big|_{x=X,y=Y} = \left.\frac{\partial f(x,y)}{\partial x}\right|_{x=X,y=Y} \cdot \vec{e}_x + \left.\frac{\partial f(x,y)}{\partial y}\right|_{x=X,y=Y} \cdot \vec{e}_y \qquad (3.80)$$

where $\vec{e}_x$ and $\vec{e}_y$ are the unit vectors of the corresponding axes.

A numerical search for a function minimum is as follows. We start from an arbitrary point (x_0,y_0), and in its vicinity we move in the direction opposite to the gradient vector.

We "arrive" at a point (x_1,y_1) such that

$$\overrightarrow{(x_1-x_0;y_1-y_0)} = -\mu_0 \,\text{grad}f(x,y)\big|_{x=x_0,y=y_0}, \qquad (3.81)$$

which with account for (3.80) leads to

$$x_1 = x_0 - \mu_0 \left.\frac{\partial f(x,y)}{\partial x}\right|_{x=x_0,y=y_0}, \quad y_1 = y_0 - \mu_0 \left.\frac{\partial f(x,y)}{\partial y}\right|_{x=x_0,y=y_0}, \tag{3.82}$$

μ_0 is a positive number, the choice of such numbers is described in what follows. We moved in the direction of the fastest function decrease, so

$$f(x_1,y_1) < f(x_0,y_0). \tag{3.83}$$

Then, similarly to the above-mentioned procedure, we move from the point (x_1,y_1) in its vicinity in the direction opposite to the gradient vector. We "arrive" at the point (x_2,y_2) which is as follows:

$$\begin{gathered}\overrightarrow{(x_2 - x_1; y_2 - y_1)} = -\mu_1 \operatorname{grad} f(x,y)\big|_{x=x_1,y=y_1}, \quad \mu_1 > 0, \\ x_2 = x_1 - \mu_1 \left.\frac{\partial f(x,y)}{\partial x}\right|_{x=x_1,y=y_1}, \quad y_2 = y_1 - \mu_1 \left.\frac{\partial f(x,y)}{\partial y}\right|_{x=x_1,y=y_1},\end{gathered} \tag{3.84}$$

and we have

$$f(x_2,y_2) < f(x_1,y_1) < f(x_0,y_0), \tag{3.85}$$

and so on. At the nth step we move from the point (x_n,y_n) to the point (x_{n+1},y_{n+1})

$$\begin{gathered}\overrightarrow{(x_{n+1} - x_n; y_{n+1} - y_n)} = -\mu_n \operatorname{grad} f(x,y)\big|_{x=x_n,y=y_n}, \quad \mu_n > 0, \\ x_{n+1} = x_n - \mu_n \left.\frac{\partial f(x,y)}{\partial x}\right|_{x=x_n,y=y_n}, \quad y_{n+1} = y_n - \mu_n \left.\frac{\partial f(x,y)}{\partial y}\right|_{x=x_n,y=y_n}, \\ f(x_{n+1},y_{n+1}) < f(x_n,y_n) < \ldots < f(x_0,y_0),\end{gathered} \tag{3.86}$$

it should be stressed that in such a numeration the movement from (x_0,y_0) to (x_1,y_1) is the zeroth step of the algorithm. The algorithm stops if the following condition is satisfied:

$$\left| f(x_{n+1},y_{n+1}) - f(x_n,y_n) \right| < \varepsilon \tag{3.87}$$

where ε is a given algorithm accuracy. The corresponding minimal value of the function is $f_{\min} = f(x_{n+1},y_{n+1})$.

Let we have a function of N variables $f(x^1,x^2,\ldots,x^N)$, the superscript here is just a variable number, not a power. The algorithm is similar to the case where

$N=2$. Sowe start from a point $\left(x_0^1, x_0^2, ..., x_0^N\right)$ and at each step we move from the point $\left(x_n^1, x_n^2, ..., x_n^N\right)$ to the point $\left(x_{n+1}^1, x_{n+1}^2, ..., x_{n+1}^N\right)$ as follows:

$$x_{n+1}^j = x_n^j - \mu_n \left.\frac{\partial f\left(x^0, x^1, ..., x^N\right)}{\partial x^j}\right|_{x^i = x_n^i}, \quad i, j = 1, 2, ..., N, \quad \mu_n > 0, \tag{3.88}$$

$$f\left(x_{n+1}^0, x_{n+1}^1, ..., x_{n+1}^N\right) < f\left(x_n^0, x_n^1, ..., x_n^N\right) < ... < f\left(x_0^0, x_0^1, ..., x_0^N\right),$$

and the algorithm stops if the following condition is satisfied:

$$\left| f\left(x_{n+1}^0, x_{n+1}^1, ..., x_{n+1}^N\right) - f\left(x_n^0, x_n^1, ..., x_n^N\right)\right| < \varepsilon, \tag{3.89}$$

ε is a given algorithm accuracy and the calculated minimal value of the function is

$$f_{\min} = f\left(x_{n+1}^0, x_{n+1}^1, ..., x_{n+1}^N\right).$$

Now let us consider the choice of the numbers μ_n. By trial and error they are chosen in order to make the algorithm converge as fast as possible. Some popular choices are:

1. Constant step method $\mu_0 = \mu_1 = \mu_2 = ... = \text{const}$

2. Fractional step method. By trial and error, constants $\alpha \in (0,1)$ and $\delta \in (0,1)$ are chosen. The nth step is as follows. Initially, μ_n is specified as μ_0. Then the following condition is checked:

$$f\left(x_n^j - \mu_n \left.\frac{\partial f}{\partial x^j}\right|_{x^i = x_n^i}\right) \le f\left(x_n^j\right) - \alpha\mu_n \cdot \sum_{j=1}^{N}\left(\left.\frac{\partial f}{\partial x^j}\right|_{x^i = x_n^i}\right)^2. \tag{3.90}$$

If (3.90) is valid. then μ_n is chosen as $\mu_n = \mu_0$. If (3.90) is not valid, then μ_n is specified as $\mu_0\delta$, and (3.90) is checked again. If it is valid, then μ_n is chosen as $\mu_n = \mu_0\delta$. If it is not valid, then μ_n is specified as $\mu_0\delta^2$ and so on. It should be stressed that the above-mentioned constants are often chosen as $\alpha = 0.5$, $\delta = 0.95$.

3. Method of quickest descent. μ_n is chosen in order to minimize the expression

$$f\left(x_n^j - \mu_n \left.\frac{\partial f}{\partial x^j}\right|_{x^i = x_n^i}\right) \to \min. \tag{3.91}$$

Let us consider the discrete Kolmogorov–Wiener problem as an example of the application of the gradient method to the filtration theory. The problem is as follows. Let the filter input be a stationary ergodic signal

$$x(t) = s(t) + n(t) \tag{3.92}$$

where $s(t)$ is a useful signal and $n(t)$ is a noise. The input signal is defined at discrete instants $t = 0,1,2,...,N$. The output signal

$$y(t) = \sum_{\tau=0}^{N} h(\tau) x(t-\tau) \tag{3.93}$$

is sought such that the mean-square deviation is a minimum

$$\left\langle \left(s(t) - y(t) \right)^2 \right\rangle_t \to \min, \tag{3.94}$$

the corresponding weight coefficients $h(j)$, $j = \overline{1,N}$ are to be found. The correlation functions $R_x(t)$ and $R_{sx}(t)$ are supposed to be given. As is known, in order to obtain the weight coefficients, we should solve the linear equation set (2.38). An alternative method of solution is the gradient method.

From (3.93) and (3.94) we have

$$\begin{aligned} &\left\langle \left(s(t) - y(t) \right)^2 \right\rangle_t = \left\langle s^2(t) \right\rangle_t - 2\left\langle s(t) y(t) \right\rangle_t + \left\langle y^2(t) \right\rangle_t = \\ &= \left\langle s^2(t) \right\rangle_t - 2\sum_{\tau=0}^{N} h(\tau) \left\langle s(t) x(t-\tau) \right\rangle_t + + \sum_{\tau=0}^{N} \sum_{\tau'=0}^{N} h(\tau) h(\tau') \left\langle x(t-\tau) x(t-\tau') \right\rangle_t . \end{aligned} \tag{3.95}$$

On the basis of (1.23) we have

$$R_s(\tau) = \left\langle s(t+\tau) s(t) \right\rangle_t \Rightarrow \left\langle s^2(t) \right\rangle_t = R_s(\tau = 0) = \text{const}, \tag{3.96}$$

from (1.36) and (1.39) we obtain

$$R_{xs}(-\tau) = \left\langle x(t-\tau) s(t) \right\rangle_t = R_{sx}(\tau), \tag{3.97}$$

and similarly to (2.23) we have

$$\left\langle x(t-\tau) x(t-\tau') \right\rangle_t = R_x(\tau - \tau'), \tag{3.98}$$

So on the basis of (3.95) – (3.98) we see that the following function of N variables should be minimized:

$$f\left(h^0,h^1,...,h^N\right)=-2\sum_{\tau=0}^{N}h^\tau R_{sx}\left(\tau\right)+\sum_{\tau=0}^{N}\sum_{\tau'=0}^{N}h^\tau h^{\tau'}R_{xx}\left(\tau-\tau'\right)\to\min,\qquad(3.99)$$
$$h^j=h\left(j\right).$$

We can use a gradient method in order to minimize the function (3.99). For example, let us use the method of quickest descent. It should be stressed that in the framework of the problem under consideration the numbers μ_n can be found analytically! Let us rewrite (3.99) in matrix form:

$$f\left(h\right)=-2h^TR_{sx}+h^TR_xh\qquad(3.100)$$

where the superscript T denotes transposition and

$$h=\begin{pmatrix}h^0\\h^1\\\vdots\\h^N\end{pmatrix},\ R_{sx}=\begin{pmatrix}R_{sx}\left(0\right)\\R_{sx}\left(1\right)\\\vdots\\R_{sx}\left(N\right)\end{pmatrix},$$
$$R_x=\begin{pmatrix}R_x\left(0\right)&R_x\left(1\right)&R_x\left(2\right)&...&R_x\left(N\right)\\R_x\left(1\right)&R_x\left(0\right)&R_x\left(1\right)&...&R_x\left(N-1\right)\\R_x\left(2\right)&R_x\left(1\right)&R_x\left(0\right)&...&R_x\left(N-2\right)\\\vdots&\vdots&\vdots&\vdots&\vdots\\R_x\left(N\right)&R_x\left(N-1\right)&R_x\left(N-2\right)&...&R_x\left(0\right)\end{pmatrix},\qquad(3.101)$$

here the fact

$$R_x\left(\tau\right)=R_x\left(-\tau\right)\qquad(3.102)$$

is used. The gradient vector is the column vector with the components

$$\operatorname{grad}_j f\left(h\right)=\frac{\partial f\left(h^0,h^1,...,h^N\right)}{\partial h^j}.\qquad(3.103)$$

On the basis of (3.99) we have

$$\begin{aligned}\operatorname{grad}_j f\left(h\right)&=-2\sum_{\tau=0}^{N}\frac{\partial h^\tau}{\partial h^j}R_{sx}\left(\tau\right)+\sum_{\tau=0}^{N}\sum_{\tau'=0}^{N}\frac{\partial h^\tau h^{\tau'}}{\partial h^j}R_x\left(\tau-\tau'\right)=\\&=-2\sum_{\tau=0}^{N}\delta_{\tau j}R_{sx}\left(\tau\right)+\sum_{\tau=0}^{N}\sum_{\tau'=0}^{N}\left(\delta_{\tau j}h^{\tau'}+\delta_{\tau' j}h^\tau\right)R_x\left(\tau-\tau'\right)=\\&=-2\sum_{\tau=0}^{N}\delta_{\tau j}R_{sx}\left(\tau\right)+\sum_{\tau=0}^{N}\sum_{\tau'=0}^{N}\delta_{\tau j}h^{\tau'}R_x\left(\tau-\tau'\right)+\sum_{\tau=0}^{N}\sum_{\tau'=0}^{N}\delta_{\tau' j}h^\tau R_x\left(\tau-\tau'\right),\end{aligned}\qquad(3.104)$$

$$\delta_{\tau j} = \begin{cases} 1, \tau = j \\ 0, \tau \neq j \end{cases}.$$

On the basis of (3.102) we have

$$\sum_{\tau=0}^{N}\sum_{\tau'=0}^{N}\delta_{\tau' j}h^{\tau}R_x(\tau-\tau') = \sum_{\tau=0}^{N}\sum_{\tau'=0}^{N}\delta_{\tau j}h^{\tau'}R_x(\tau'-\tau) = \sum_{\tau=0}^{N}\sum_{\tau'=0}^{N}\delta_{\tau j}h^{\tau'}R_x(\tau-\tau'), \qquad (3.105)$$

and from (3.104) and (3.105) we can obtain

$$\text{grad}_j f(h) = -2R_{sx}(j) + 2\sum_{\tau=0}^{N}R_x(\tau - j)h^{\tau}, \qquad (3.106)$$

so in matrix form

$$\text{grad}f(h) = -2R_{sx} + 2R_x \cdot h. \qquad (3.107)$$

Let us find an expression for μ_n. Condition (3.91) can be rewritten in matrix form:

$$f(h - \mu \cdot \text{grad}f(h)) \to \min, \qquad (3.108)$$

here and in what follows the subscript n is omitted for simplicity. On the basis of (3.100) we have

$$\begin{aligned} f(h - \mu \cdot \text{grad}f(h)) = -2(h - \mu \cdot \text{grad}f(h))^T R_{sx} + \\ +(h - \mu \cdot \text{grad}f(h))^T R_x (h - \mu \cdot \text{grad}f(h)), \end{aligned} \qquad (3.109)$$

which by a straightforward calculation leads to

$$\begin{aligned} f(h - \mu \cdot \text{grad}f(h)) = h^T R_x h - 2h^T R_{sx} + \\ +\mu\left(2(\text{grad}f(h))^T R_{sx} - (\text{grad}f(h))^T R_x h - h^T R_x \text{grad}f(h)\right) + \\ +\mu^2 \cdot (\text{grad}f(h))^T R_x \text{grad}f(h). \end{aligned} \qquad (3.110)$$

As can be seen from (3.101), R_x is a symmetric matrix, so

$$R_x = R_x^T, \ \left((\text{grad}f(h))^T R_x h\right)^T = h^T R_x^T \text{grad}f(h) = h^T R_x \text{grad}f(h). \qquad (3.111)$$

Obviously, $h^T R_x \text{grad}f(h)$ is a number (1×1 matrix), so on the basis of (3.111) we have

$$R_x = R_x^T, \ (\text{grad}f(h))^T R_x h = \left((\text{grad}f(h))^T R_x h\right)^T = h^T R_x \text{grad}f(h), \qquad (3.112)$$

which with account for (3.110) leads to

$$f(h - \mu \cdot \text{grad}f(h)) = h^T R_x h - 2h^T R_{sx} + \qquad (3.113)$$

$$+2\mu\left(\left(\text{grad}f(h)\right)^T R_{sx} - h^T R_x \text{grad}f(h)\right) +$$
$$+\mu^2 \cdot \left(\text{grad}f(h)\right)^T R_x \text{grad}f(h) \to \min,$$

so

$$\frac{df\left(h - \mu \cdot \text{grad}f(h)\right)}{d\mu} = 0 \Rightarrow \mu = \frac{h^T \cdot R_x \cdot \text{grad}f(h) - \left(\text{grad}f(h)\right)^T \cdot R_{sx}}{\left(\text{grad}f(h)\right)^T \cdot R_x \cdot \text{grad}f(h)}, \qquad (3.114)$$

it should be mentioned that both the numerator and the denominator of (3.114) are numbers rather than matrices.

To summarize the above-mentioned, the discrete Kolmogorov–Wiener problem can be reduced to a search for a minimum of the multivariable function (3.99). Such a problem can be solved on the basis of a gradient method. In the case of the method of quickest descent, the numbers μ_n can be analytically expressed as follows:

$$\mu_n = \left.\frac{h^T \cdot R_x \cdot \text{grad}f(h) - \left(\text{grad}f(h)\right)^T \cdot R_{sx}}{\left(\text{grad}f(h)\right)^T \cdot R_x \cdot \text{grad}f(h)}\right|_{h^j = h_n^j}. \qquad (3.115)$$

At the end we should mention that gradient methods may be applied to a search for a maximum of a multivariable function. In such a case, we should move in the direction that coincides with the gradient vector rather than in the direction opposite to the gradient vector.

3.10 Discrete digital filter adaptation algorithms

Before considering the actual adaptation algorithms, it is necessary to determine the optimal filter parameters to which these algorithms should strive. The approach to the optimal filtering problem can be both statistical and deterministic. First, consider the statistical approach.

Let the input discrete random signal $x(k)$ be processed by a non-recursive discrete filter of order N, the coefficients of which can be represented by the column vector $w = \left(w_0, w_1, .., w_N\right)^t$. The output of the filter is

$$y(k) = u^t(k)w \qquad (3.116)$$

where $u(k) = \left(x(k), x(k-1), ..., x(k-N)\right)^t$ is the column vector of the filter's delay line at the *k*-th step.

In addition, there is an exemplary (also random) signal *d*(*k*). The sample playback error is equal to

$$e(k) = d(k) - y(k) = d(k) - u^t(k)w. \qquad (3.117)$$

It is necessary to find such filter coefficients *w*, which ensure the maximum proximity of the output signal of the filter to the exemplary one, that is, minimize the error *e*(*k*). Since *e*(*k*) is also a random process, it is reasonable to take the average square as a measure of its value. Thus, the optimized functionality looks like this:

$$J(w) = \left\langle e^2(k) \right\rangle \to \min$$

Error square equals

$$e^2(k) = \left(d(k) - u^t(k)w\right)^2 = d^2(k) - 2d(k)u^t(k)w + w^t u(k)u^t(k)w$$

Statistically averaging this expression, we get the following:

$$J(w) = \left\langle e^2(k) \right\rangle = \left\langle d^2(k) \right\rangle - 2\left\langle d(k)u^t(k) \right\rangle w + w^t \left\langle u(k)u^t(k) \right\rangle w. \qquad (3.118)$$

The averaged values included in the resulting formula have the following meaning:

- $\left\langle d^2(k) \right\rangle = \sigma_d^2$ – the average square of the exemplary signal;

- $\left\langle d(k)u^t(k) \right\rangle = p^t$ – transposed column vector *p* of mutual correlations between the *k*-th sample of the sample signal and the contents of the filter delay line. If the considered random processes *x*(*t*) and *d*(*k*) are jointly stationary, the vector of mutual correlations does not depend on the number of step *k*;

- $\left\langle u(k)u^t(k) \right\rangle = R$ - the correlation matrix of the signal having a size (*N* + 1) × (*N* + 1). For a stationary random process, the correlation matrix has the form of a Toeplitz matrix, that is, the same values are on its diagonals:

$$R = \begin{pmatrix} R_x(0) & R_x(1) & R_x(2) & \cdots & R_x(N) \\ R_x(1) & R_x(0) & R_x(1) & \cdots & R_x(N-1) \\ R_x(2) & R_x(1) & R_x(0) & \cdots & R_x(N-2) \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ R_x(N) & R_x(N-1) & R_x(N-2) & \cdots & R_x(0) \end{pmatrix}.$$

Taking into account the introduced notation, (3.118) takes the following form:

$$J(w) = \sigma_d^2 - 2p^t w + w^t R w. \tag{3.119}$$

This expression is a quadratic form in w, and therefore, with a non-singular matrix R has a unique minimum, for finding which it is necessary to equate the gradient vector to zero:

$$\text{grad}J(w) = -2p + 2Rw = 0.$$

From here we get the desired solution for optimal filter coefficients:

$$w = R^{-1} p. \tag{3.120}$$

Such a filter is called a Wiener filter. Substituting (3.120) into (3.119) gives the minimum achievable error signal variance:

$$\left\langle e^2(k)_{\min} \right\rangle = \sigma_d^2 - p^t R^{-1} p. \tag{3.121}$$

It is also easy to show that $\langle e(k)x(k) \rangle = 0$, that is, that the error signal for the Wiener filter is uncorrelated with the input and output signals of the filter.

3.10.1 LMS Algorithm

Least Mean Square (LMS) algorithm belongs to the class of algorithms for stochastic gradient coordinate search for the minimum of the target function.

It is one of the most common adaptive algorithms based on finding the minimum of the target function (3.118) using the steepest descent method. When using this optimization method, the filter coefficient vector $w(k)$ should be recursively updated as follows:

$$w(k+1) = w(k) - \frac{\mu}{2}\text{grad}J\left(w(k)\right) = w(k) + \mu p - \mu R w(k), \tag{3.122}$$

where μ is a positive coefficient called step size. A detailed analysis of the convergence of this process shows that the algorithm converges if $0 < \mu < 2/\lambda_{\max}$, where $\lambda_{\max}$ is the maximum eigenvalue of the correlation matrix R. The convergence rate depends on the spread of the eigenvalues of the correlation matrix R - the smaller the ratio $\lambda_{\max} / \lambda_{\min}$, the faster the iterative process converges.

However, to calculate the gradient, it is necessary to know the values of the matrix R and the vector p. In practice, only estimates of these values obtained frominput data can be available. The simplest estimates of this type are the instantaneous values of the correlation matrix and the vector of mutual correlations obtained without any averaging:

$$\begin{aligned} w(k+1) &= w(k) + \mu d(k)u(k) - \mu u(k)u^t(k)w(k) = \\ &= w(k) + \mu u(k)(d(k) - u^t(k)w(k)). \end{aligned} \tag{3.123}$$

The expression in brackets, according to (3.117), is the difference between the exemplary signal and the filter output signal at the k-th step, that is, the filtering error $e(k)$. With this in mind, the expression for recursively updating filter coefficients is very simple:

$$w(k+1) = w(k) + \mu e(k)u(k). \tag{3.124}$$

The adaptive filtering algorithm based on formula (3.124) is called LMS (Least Mean Square, the least squares method). An analysis of the convergence of the LMS algorithm shows that the upper limit for step size μ in this case is smaller than when using true gradient values. This border is approximately equal to

$$\mu_{\max} \approx \frac{2}{\sum_k \lambda_k} = \frac{2}{\mathrm{Tr}(R)} = \frac{2}{(N+1)\sigma_x^2} \tag{3.125}$$

where λ_k are the eigenvalues of the correlation matrix R, and σ_x^2 is the average square of the input filter signal. The main advantage of the LMS algorithm is extreme computational simplicity - to adjust the filter coefficients at each step, you need to perform $N+1$ pairs of multiplication-addition operations. The price for simplicity is slow convergence and increased, as compared to the minimum attainable value (3.121), error variance in steady state - filter coefficients always fluctuate around optimal values (3.120), which increases the level of output noise.

There are a large number of modifications of the LMS algorithm aimed at accelerating convergence or reducing the number of arithmetic operations. Acceleration of convergence can be achieved by improving the gradient estimation used, as well as by converting the input signal to make its samples uncorrelated.

Reducing the computational complexity can be achieved, in particular, by using in (3.124 only the signs of the error signal and the contents of the filter's delay line. This allows one to completely get rid of multiplication operations when updating filter coefficients. In general, it should be noted that the requirements for accelerating convergence and reducing computational costs are contradictory.

3.10.2 Deterministic optimization problem

Considering the statistical optimization problem, we considered the input signal as *a random process* and minimized the *mean square* of the exemplary signal reproduction error. However, a different approach is possible without using statistical methods.

Let, as before, the sequence consisting of K samples $x(k)$ be subjected to processing. The coefficients of a non-recursive filter form a column vector w, and the samples of the exemplary signal are equal to $d(k)$. The output signal of the filter is determined by formula (3.116), and the error of reproduction of the sample signal is determined by formula (3.117). Now the optimization problem is formulated as follows: one needs to find filter coefficients w such that the error norm of the exemplary signal reproduction is minimal:

$$J(w) = \sum_{k=0}^{K-1} |e(k)|^2 \to \min. \qquad (3.126)$$

To solve the problem in expressions (3.116) and (3.117), it is necessary to go to the matrix notation along the k coordinate, obtaining the formulas for the column vectors of the output signal y and for the playback error of the input signal e:

$$y = U^t w, \quad e = d - U^t w. \qquad (3.127)$$

Here d is the column vector of samples of the exemplary signal, and U is the matrix, the columns of which represent the contents of the filter delay line at different times:

$$U = \big(u(0), u(1), \ldots, u(K-1)\big).$$

Expression (3.126) for the error norm can be rewritten in a matrix form as follows:

$$J(w) = e^t e \to \min. \qquad (3.128)$$

Substituting (3.127) into (3.128), we have

$$J(w) = (d - U^t w)^t (d - U^t w) = d^t d - w^t U d - d^t U^t w + w^t U U^t w.$$

To find the minimum, it is necessary to calculate the gradient of this functional and equate it to zero:

$$\text{grad}J(w) = -2Ud + 2UU^t w = 0.$$

From here, the desired optimal solution is easily obtained:

$$w = (UU^t)^{-1} Ud. \qquad (3.129)$$

In formula (3.129), there is a close relationship with formula (3.120), which describes the statistical Wiener filter, which is statistically optimal. Indeed, if we take into account that $(UU^t)^{-1} / K$ gives an estimate of the correlation matrix of the signal obtained by a single signal implementation by temporal averaging, and Ud / K is a similar estimate of the mutual correlations between the exemplary signal and the contents of the filter delay line, then formulas (3.129) and (3.120) match.

3.10.3 RLS Algorithm

In the process of receiving a signal, at each successive step, the coefficients of the filter can be recalculated directly according to formula (3.129), but this is associated with unnecessarily large computational costs. Indeed, the size of the matrix U is constantly increasing and, moreover, it is necessary each time to re-calculate the inverse matrix $(UU^t)^{-1}$.

It is possible to reduce computational costs if we note that at each step only one new column is added to the matrix U, and one new element is added to the vector d. This makes it possible to organize calculations *recursively.* The corresponding algorithm is called the Recursive Least Square Method (RLS).

Without giving a detailed derivation of the formulas describing the RLS algorithm, here we present only the main idea of the method. When using the RLS algorithm, the estimate of the inverse correlation matrix $P = (UU^t)^{-1}$ is recursively updated, and the derivation of the formulas is based on the following matrix identity:

$$(A+BCD)^{-1} = A^{-1} - A^{-1}B(C^{-1} + DA^{-1}B)^{-1}DA^{-1} \qquad (3.130)$$

where A and C are square non-singular matrices (not necessarily of the same size), and B and D are matrices of compatible sizes.

The use of formula (3.130) for the recursive update of the inverse correlation matrix P in combination with the original formula (3.129) for calculating the optimal filter coefficients gives the following sequence of steps of the adaptive RLS algorithm.

1 When new input data $u(k)$ are received, the signal is filtered using the current filter coefficients $w(k-1)$ and the exemplary error value is reproduced:

$$y(k) = u^t(k)w(k-1),\ e(k) = d(k) - y(k).$$

2 Calculate the gain column vector (it should be noted that the fraction denominator in the following two formulas is a scalar, not a matrix):

$$K(k) = \frac{P(k-1)u(\mathrm{k})}{1 + u^t(k)P(k-1)u(k)}. \qquad (3.131)$$

3 The estimation of the inverse correlation matrix of the signal is updated:

$$P(k) = P(k-1) - K(k)u^t(k)P(k-1). \qquad (3.132)$$

4 Finally, the filter coefficients are updated:

$$w(k) = w(k-1) + K(k)e(k).$$

The initial value of the vector w is usually taken to be zero, and the diagonal matrix of the form CI/σ_x^2 is used as the initial estimate of the matrix P, where $C >> 1$ (C is recommended ≥ 100).

In formulas (3.126) and (3.128), the same weight is attached to the error values on all time ticks. As a result, if the statistical properties of the input signal change with time, this leads to a deterioration in the quality of filtering. To enable the filter to track a non-stationary input signal, one can apply in (3.126) an exponential

forgetting, at which the weight of past values of the error signal exponentially decreases:

$$J(w) = \sum_{k=0}^{K-1} \lambda^{K-1-k} \left| e(k) \right|^2, \; 0 \le \lambda < 1.$$

When using exponential forgetting, formulas (3.131) and (3.132) take the following form:

$$K(k) = \frac{P(k-1)u(k)}{\lambda + u^t(k)P(k-1)u(k)}, \; P(k) = \frac{1}{\lambda}\left(P(k-1) - K(k)u^t(k)P(k-1) \right).$$

The main advantage of the RLS algorithm is fast convergence. However, this is achieved due to a significantly higher (compared to the LMS algorithm) computational complexity. With optimal computation, updating the filter coefficients at each cycle requires $(2.5N^2 + 4N)$ pairs of multiplication – addition operations.

Chapter 4. Practical applications of adaptive filters

The basic configuration of an adaptive filter, operating in the discrete-time domain k, is illustrated in Figure 4.1. In such a scheme, the input signal is denoted

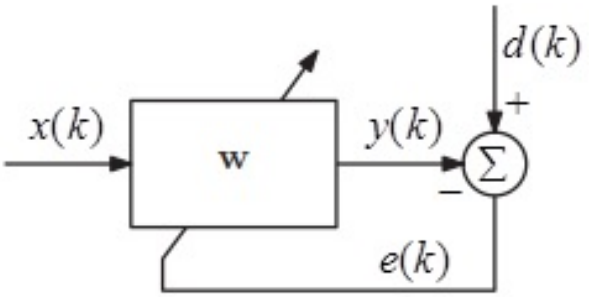


Figure. 4.1 – Basic block diagram of an adaptive filter

by $x(k)$, the reference signal $d(k)$ represents the desired output signal (that usually includes some noise component), $y(k)$ is the output of the adaptive filter, and the error signal is defined as $e(k) = d(k) - y(k)$.

The error signal is used by the adaptation algorithm to update the adaptive filter coefficient vector $w(k)$ according to some performance criterion. In general, the whole adaptation process aims at minimizing some metric of the error signal, forcing the adaptive filter output signal to approximate the reference signal in a statistical sense.

It is interesting to notice how this basic configuration fits perfectly in several practical applications such as system identification, interference canceling, channel equalization, and signal prediction, which are detailed as follows.

For instance, Figure 4.2 depicts a typical system identification configuration, where w_o is an ideal coefficient vector of a_n unknown plant, whose output is represented by $y_o(k)$, and $n(k)$ denotes the observation or measurement noise. In this setup, the plant and the adaptive filter receive the same input signal. After convergence, the output signals of both systems become similar, and consequently

the adaptive transfer function becomes a good model for the input–output relationship of the plant.

Another application of an adaptive filter is interference canceling or signal enhancement represented in Figure 4.3. In this problem, a signal of interest $s(k)$ is corrupted by a noise component $n(k)$. A cleaner version of $s(k)$ is desired but cannot be obtained directly in practice. The noisy signal, $s(k) + n(k)$, is then employed as the reference signal for the adaptive filter, whose input must be another version $\hat{n}(k)$, of the noise signal, strongly correlated to $n(k)$. The adaptive mechanism adjusts the filter coefficients in such a manner that the filter output $y(k)$ approximates $n(k)$, thus forcing the error signal $e(k)$ to resemble signal $s(k)$.

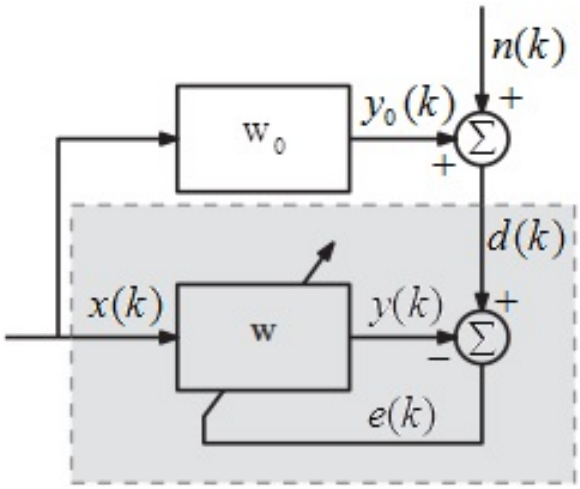


Figure 4.2 – System identification configuration of an adaptive filter: The adaptive coefficient w vector estimates the unknown system coefficient vector w_o.

In practical communications systems, a transmitted signal can be heavily distorted by the transmission channel. One may attempt to recover the original signal by employing an adaptive filter in the channel equalization configuration, as depicted in Figure 4.4. In such a framework, a training sequence $s(k)$ known by the receiver is sent via a given channel generating a distorted signal. The same sequence $s(k)$, after a proper time shift to compensate for transmission delays, is used as a reference signal in the receiver for the adaptive filter, whose input is the distorted signal. When the error function approximates zero, the output signal $y(k)$ resembles the transmitted

signal $s(k)$, indicating that the adaptive filter is compensating for the channel distortions. After this training process, the desired information can be sent through the channel, which is properly equalized by the adaptive filter.

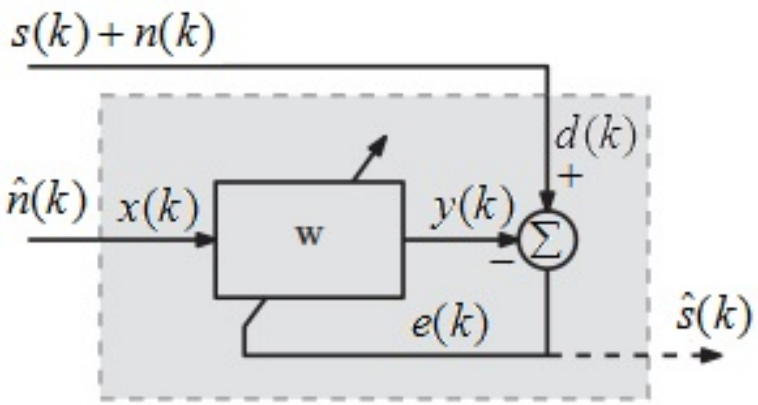


Figure 4.3 – Interference cancelation configuration of an adaptive filter: The error signal $e(k)$ approximates the desired signal component $s(k)$ if $n(k)$ and $\hat{n}(k)$ are correlated.

The adaptive predictor configuration is depicted in Figure 4.5. In this case, the adaptive filter input signal $x(k)$ is a delayed version of the reference signal $d(k)$.

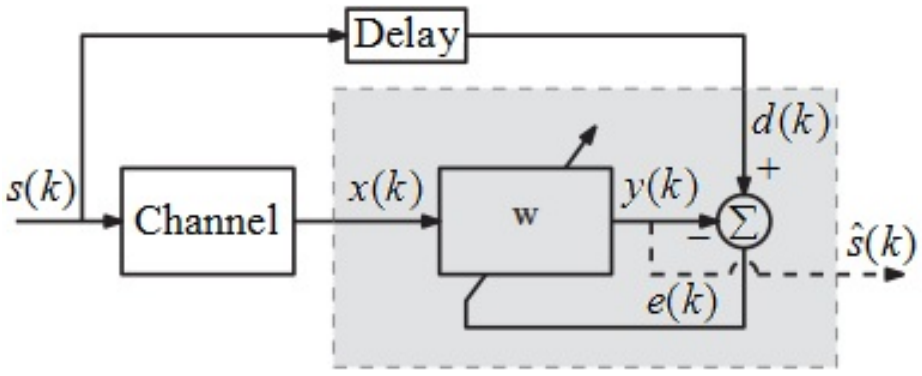


Figure 4.4 – Channel equalization configuration of an adaptive filter: The output signal $y(k)$ estimates the transmitted signal $s(k)$

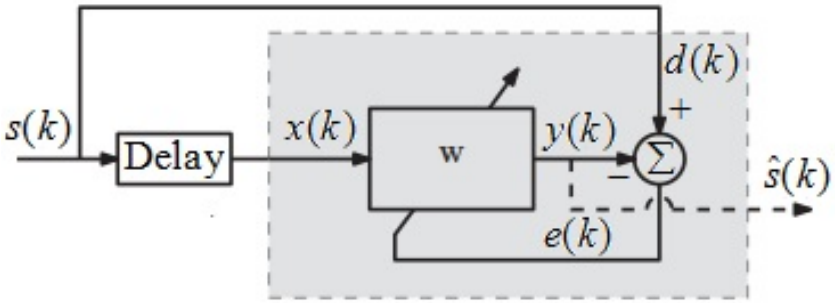


Figure 4.5 – Predictor configuration of an adaptive filter: The output signal *y*(k) estimates the present input sample *s*(*k*) based on past values of this same signal

Therefore, when the adaptive filter output y(k) approximates the reference, the adaptive filter operates as a predictor system.

From the discussion so far, one observes that the reference signal, through the definition of the error signal, acts as a general guide for the entire adaptation process. The four configurations illustrated above indicate how one can determine the desired output signal in several practical situations. In all cases, one can clearly identify the adaptive filter block given in Figure 4.1. To completely characterize this common basic cell, three main aspects must be defined:

1. Adaptive filter structure: This book will focus on the adaptive transversal FIR structure, whose input–output relationship is described by

$$y(k) = w_0 x(k) + w_1 x(k-1) + \ldots + w_N x(k-N) = \sum_{i=0}^{N} w_i x(k-i) = w^t x(k)$$

where N is the filter order and $x(k)$ and w are vectors composed by the input-signal samples and the filter coefficients, respectively; that is

$$x(k) = \left(x(k), x(k-1), \ldots, x(k-N)\right)^t, \quad w = \left(w_0, w_1, \ldots, w_N\right)^t.$$

In cases of complex implementations, the output signal is represented as $w^H x(k)$, where the superscript H denotes the Hermitian operator (transpose and complex conjugate).

2. Error metric: As mentioned before, the adaptation algorithms adjust the adaptive filter coefficients in an attempt to minimize a given error norm. Different metrics yield adaptation processes with quite distinct characteristics.

3. Adaptation algorithm: Several optimization procedures can be employed to adjust the filter coefficients, including, for instance, the least mean-square (LMS) and its normalized version, the data-reusing (DR) including the affine projection (AP), and the recursive least-squares (RLS) algorithms.

Let us consider in more detail the practical application of adaptive filters.

4.1 Using RLS Adaptive Filters for System Identification

Adaptive filters are used for non-stationary signals and environments or in applications where a sample-by-sample adaptation of a process or a low processing delay is required. The characteristics of digital filters can easily be changed by modifying the filter coefficients. The basic concept of an adaptive filter is shown in Figure 4.6.

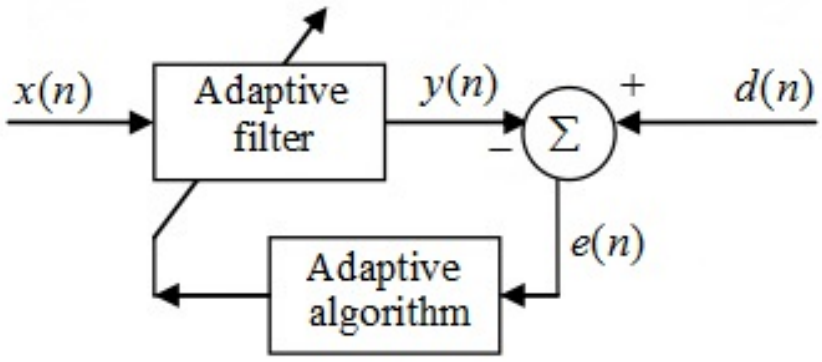


Figure 4.6 – Adaptive filter

The objective is to filter the input signal, $x(n)$, with an adaptive filter in such a manner that it matches the desired signal, $d(n)$. The desired signal, $d(n)$, is subtracted

from the filtered signal, $y(n)$, to generate an error signal. The error signal drives an adaptive algorithm that generates the filter coefficients in a manner that minimizes the error signal. The least-mean-square (LMS) or recursive-least-squares (RLS) adaptive filter are two of the most popular ones.

Identifying an unknown system has been a central issue in various application areas such as control, channel equalization, echo cancellation in communication networks and teleconferencing etc. Identification is the procedure of specifying the unknown model in terms of the available experimental evidence, that is, a set of measurements of the input output desired response signals and an appropriately error that is optimized with respect to unknown model parameters. Adaptive identification refers to a particular procedure where we learn more about the model as each new pair of measurements is received and we update the knowledge to incorporate the newly received information.

In the wide range of available adaptive algorithms, gradient descend methods, including the popular least mean squares (LMS) and recursive-least-squares (RLS) adaptive filter, are used.

4.1.1 RLS Adaptive Filter

The recursive least square error (RLS) filter is a sample-adaptive, time-update, version of the Wiener filter. For stationary signals, the RLS filter converges to the same optimal filter coefficients as the Wiener filter. For non-stationary signals, the RLS filter tracks the time variations of the process. The RLS filter has a relatively fast rate of convergence to the optimal filter coefficients. Figure 4.7 illustrates the configuration of an adaptive filter where $y(m)$, $x(m)$, $w(m) = \left(w_0(m), w_1(m), ..., w_{P-1}(m)\right)^t$ and P denote the filter input, the desired signal, the filter coefficient vector and the filter length, respectively. The filter output can be expressed as

$$\hat{x}(m) = w^t(m)y(m) \tag{4.1}$$

where $\hat{x}(m)$ is an estimate of the desired signal $x(m)$. The filter error signal is defined as

$$e(m) = x(m) - \hat{x}(m) = x(m) - w^t(m)y(m). \tag{4.2}$$

The adaptation process is based on the minimization of the mean square error criterion defined as

$$\begin{aligned} E[e^2(m)] &= E\left\{\left[x(m) - w^t(m)y(m)\right]^2\right\} = \\ &= r_{xx}(0) - 2w^t(m)r_{yx} + w^t R_{yy}(m)w(m). \end{aligned} \tag{4.3}$$

For stationary signals, the result of this minimization is given as

$$w = R_{yy}^{-1} r_{yx}\ , \tag{4.4}$$

where, R_{yy} is the autocorrelation matrix of the input signal and r_{yx} is the cross-correlation vector of the input and the target signals. For a block of N sample vectors, the correlation matrix can be written as

$$R_{yy} = Y^t Y = \sum_{m=0}^{N-1} y(m)y^t(m) \tag{4.5}$$

where $y(m) = \left(y(m),...,w(m-P)\right)^t$. Now, the sum of vector product in (4.5) can be expressed in recursive fashion as

$$R_{yy}(m) = R_{yy}(m-1) + y(m)y^t(m). \tag{4.6}$$

To introduce adaptability to the time variations of the signal statistics, the autocorrelation estimate in (4.6) can be windowed by an exponentially decaying window:

$$R_{yy}(m) = \lambda R_{yy}(m-1) + y(m)y^t(m) \tag{4.7}$$

where λ is the so-called adaptation, or forgetting factor, which lies in the range $0<\lambda\leq 1$. Similarly, the cross-correlation vector is given by

$$r_{yx} = \sum_{m=0}^{N-1} y(m)x(m)\,. \tag{4.8}$$

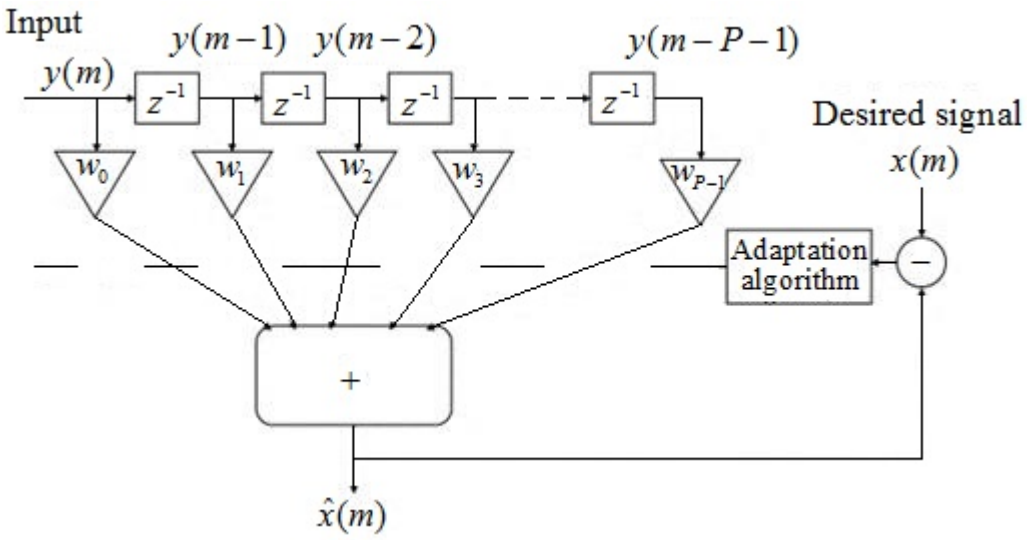


Figure 4.7 – Configuration of an adaptive filter

The sum of products in Equation (4.8) can be calculated in recursive form as

$$r_{yx}(m) = r_{yx}(m-1) + y(m)x(m)\,. \tag{4.9}$$

Again this equation can be made adaptive using an exponentially decaying forgetting factor λ:

$$r_{yx}(m) = \lambda r_{yx}(m-1) + y(m)x(m)\,. \tag{4.10}$$

For a recursive solution of the least square error Equation (4.10), we should obtain a recursive time-update formula for the inverse matrix in the form

$$R_{yy}^{-1} = R_{yy}^{-1}(m-1) + \text{Update}(m)\,. \tag{4.11}$$

4.1.2 Recursive Time-update of Filter Coefficients

The least square error filter coefficients are

$$w(m) = R_{yy}^{-1}(m)r_{yx}(m) = \Phi_{yy}(m)r_{yx}(m)\,. \tag{4.12}$$

Substituting the recursive form of the correlation vector into Equation (4.12) from Equation (4.10) yields

$$w(m) = \Phi_{yy}(m)\left[\lambda r_{yx}(m) + y(m)x(m)\right]. \tag{4.13}$$

Now substitution of $k(m)=\Phi(m)y(m)$ and the recursive form of $\Phi_{yy}(m) = \lambda^{-1}\Phi_{yy}(m-1) - \lambda^{-1}k(m)y^t(m)\Phi_{yy}(m-1)$ into Equation (4.13) yields

$$w(m) = \left[\lambda^{-1}\Phi(m-1) - \lambda^{-1}k(m)y^t(m)\Phi_{yy}(m-1)\right]\lambda r_{yx}(m-1) + k(m)x(m). \tag{4.14}$$

Substitution of $w(m–1)=\Phi(m–1)r_{yx}(m–1)$ into (4.14) yields

$$w(m) = w(m-1) - k(m)\left[x(m) - y^t(m)w(m-1)\right]. \tag{4.15}$$

This equation can be rewritten in the following form

$$w(m) = w(m-1) - k(m)e(m). \tag{4.16}$$

Equation (4.16) is a recursive time-update implementation of the least square error Wiener filter.

4.1.3 The Steepest-Descent Method

The mean square error surface with respect to the coefficients of an FIR filter is a quadratic bowl-shaped curve, with a single global minimum that corresponds to the LSE filter coefficients as shown in Figure 4.8. The steepest descent search is based on taking a number of successive downward steps in the direction of negative gradient of the error surface. The steepest-descent adaptation method can be expressed as

$$w(m+1) = w(m) + \mu\left[-\frac{\partial E(m)}{\partial w(m)}\right], \tag{4.17}$$

where, μ is the adaptation step size.

The gradient of the mean square error function is given by

$$\frac{\partial E[e^2(m)]}{\partial w(m)} = -2r_{yx} + 2R_{yy}w(m). \tag{4.18}$$

Substituting (4.18) into (4/17) yields

$$w(m+1) = w(m) + \mu\left[r_{yx} - R_{yy}w(m)\right]. \tag{4.19}$$

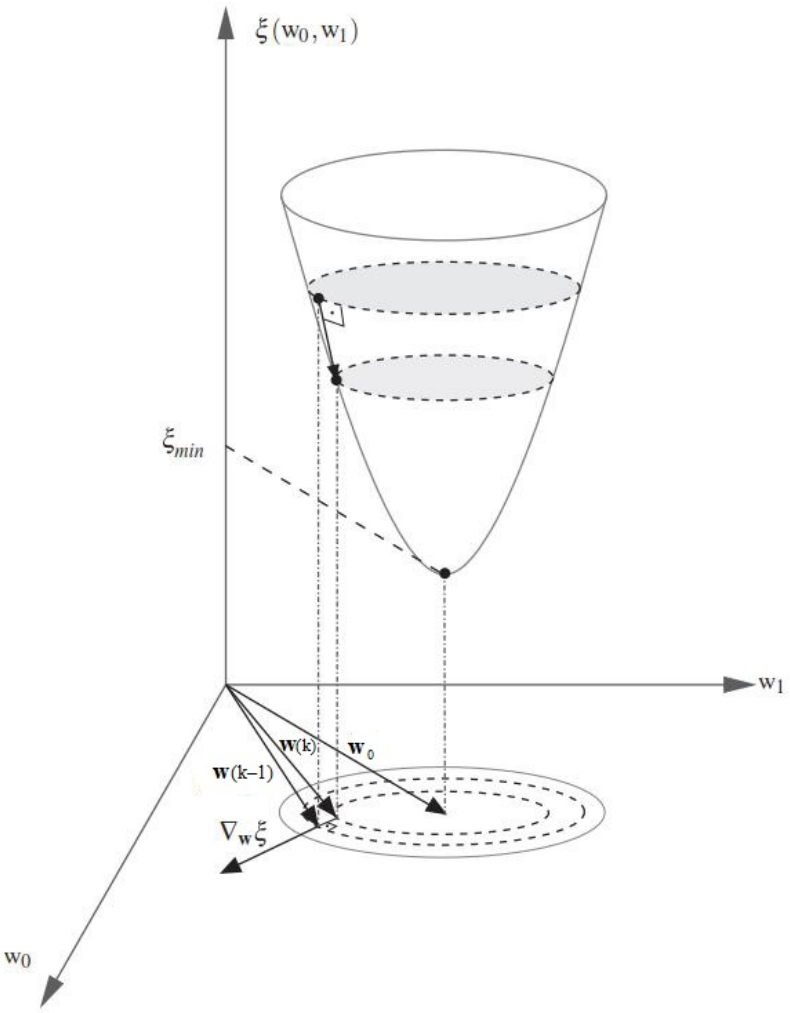


Figure 4.8 – Coefficient updating in a steepest-descent-based algorithm.

Let w_0 denote the optimal LSE filter coefficient vector, we define a filter coefficients error vector $\tilde{w}(m)$ as

$$\tilde{w}(m) = w(m) - w_0. \tag{4.20}$$

For a stationary process, the optimal LSE filter w_0 is obtained from Wiener filter, as

$$w_0 = R_{yy}^{-1} r_{yx} . \tag{4.21}$$

By substituting (4.19) into (4.20), we get

$$\tilde{w}(m+1) = \left[I - \mu R_{yy} \right] \tilde{w}(m) . \tag{4.22}$$

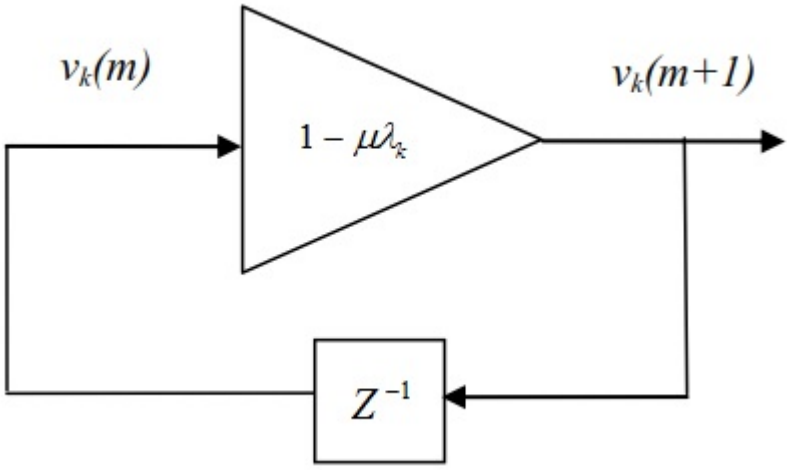


Figure 4.9 A feedback model of the variation of coefficient error with time

The parameter μ, the adaptation step size, controls the stability and the rate of convergence of the adaptive filter. Too large a value for μ causes instability; too small a value gives a low convergence rate. The correlation matrix can be expressed in terms of the matrices of eigenvectors and eigenvalues as

$$R_{yy} = Q \Lambda Q^t , \tag{4.23}$$

where, Q is an ortho-normal matrix of the eigenvectors of R_{yy}, and Λ is a diagonal matrix with its diagonal elements corresponding to the eigenvalues of R_{yy}. Substituting R_{yy} from Equation (4.23) into Equation (4.22) yields

$$\tilde{w}(m+1) = \left[I - \mu Q \Lambda Q^t \right] \tilde{w}(m) \tag{4.24}$$

Multiplying both sides of Equation (4.24) by Q^t and using the relation Q^tQ = $=QQ^t = I$ yields

$$Q^t\tilde{w}(m+1) = \left[I - \mu\Lambda\right]Q^t\tilde{w}(m). \tag{4.25}$$

Let us denote

$$v(m) = Q^t\tilde{w}(m). \tag{4.26}$$

Then

$$v(m+1) = \left[I - \mu\Lambda\right]v(m). \tag{4.27}$$

As Λ and I are both diagonal matrices, Equation (4.27) can be expressed in terms of the equations for the individual elements of the error vector $v(m)$ as

$$v_k(m+1) = \left[I - \mu\lambda_k\right]v_k(m), \tag{4.28}$$

where λ_k is the k^{th} eigenvalue of the autocorrelation matrix of the filter input $y(m)$.

4.2 Applications of Adaptive Filters

The most important driving forces behind the developments in adaptive filters throughout their history have been the wide range of applications in which such systems can be used. The major applications of adaptive filters are system identification, inverse modeling, linear prediction and feed-forward control.

4.2.1 System Identification

Figure 4.10 shows the general problem of *system identification*. In this diagram, the system enclosed by dashed lines is a "black box," meaning that the quantities inside are not observable from the outside. Inside this box is an unknown system which represents a general input-output relationship and the signal $\eta(n)$, called the *observation noise signal* because it corrupts the observations of the signal at the output of the unknown system.

Let $d'(n)$ represent the output of the unknown system with $x(n)$ as its input. Then, the desired response signal in this model is

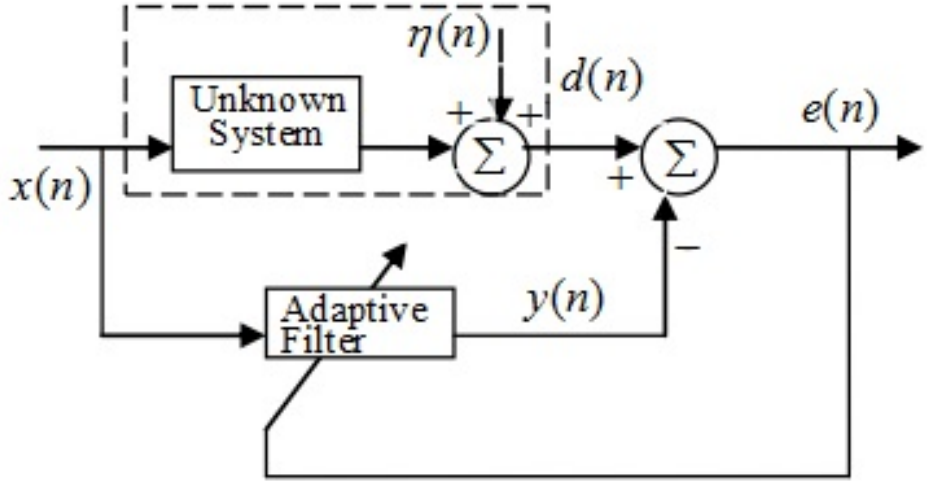


Figure 4.10 – System Identification

$$d(n) = d'(n) + \eta(n). \tag{4.29}$$

The task of the adaptive filter is to accurately represent the signal $d'(n)$ at its output. If $y(n) = d'(n)$, then the adaptive filter has accurately modeled or identified the portion of the unknown system that is driven by $x(n)$.

Let both the unknown system and the adaptive filter be FIR filters, such that

$$d(n) = W_{opt}^{t}(n)X(n) + \eta(n) \tag{4.30}$$

where, $W_{opt}^{t}(n)$ is an optimum set of filter coefficients for the unknown system at time n. In the system identification there are two major applications, one is channel identification and the other is adaptive noise cancellation.

4.2.2 Inverse Modelling

The *inverse modelling* system is shown in Figure 4.11. In this diagram, *a source signal* $s(n)$ is fed into an unknown system that produces the input signal $x(n)$ for the adaptive filter.

The output of the adaptive filter is subtracted from a desired response signal that is a delayed version of the source signal, such that

$$d(n) = s(n-\Delta) \tag{4.31}$$

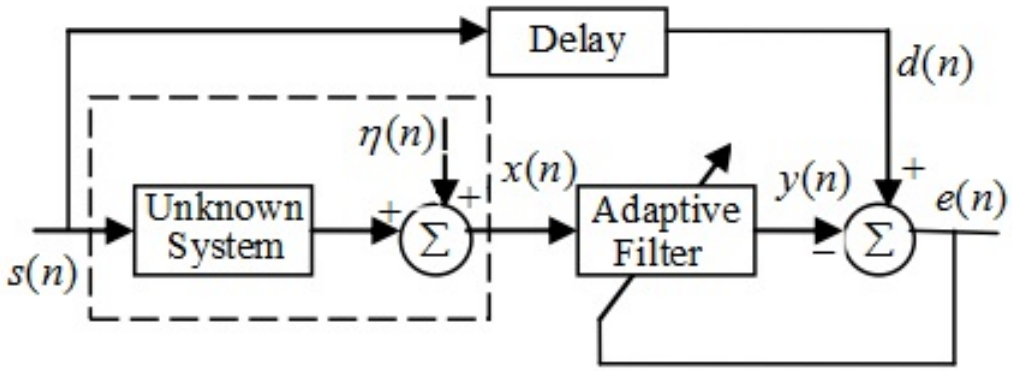


Figure 4.11 – Inverse Modeling

where Δ is a positive integer value. The goal of the adaptive filter is to adjust its characteristics so that the output signal is an accurate representation of the delayed source signal.

4.2.3 Feedforward Control

Another problem area combines elements of both the inverse modeling and system identification tasks and typifies the types of problems encountered in the area of adaptive control known as feedforward control. Figure 4.12 shows the block diagram for this system, in which the output of the adaptive filter passes through a plant before it is subtracted from the desired response to form the error signal. The plant hampers the operation of the adaptive filter by changing the amplitude and phase characteristics of the adaptive filter's output signal as represented in $e(n)$.

Thus, knowledge of the plant is generally required in order to adapt the parameters of the filter properly.

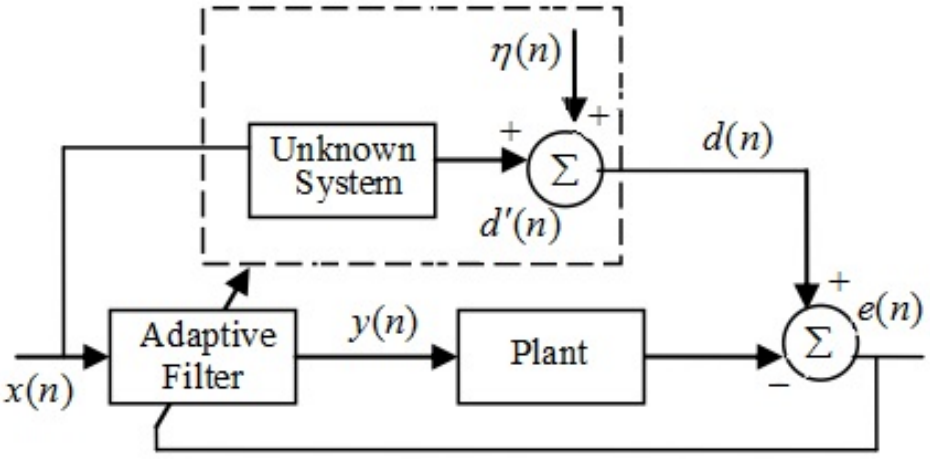


Figure 4.12 – Feed forward Control

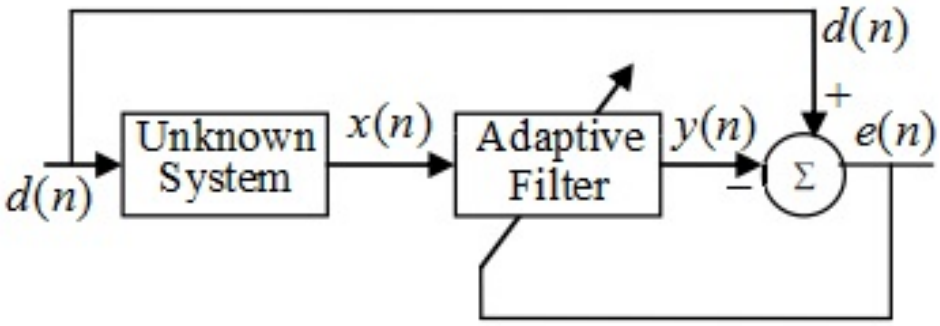


Figure 4.13 – Linear Prediction

4.2.4 Linear Prediction

A third type of adaptive filtering task is shown in Figure 4.13. In this system, the input signal $x(n)$ is derived from the desired response signal as

$$x(n) = d(n–\Delta), \tag{4.32}$$

where, Δ is an integer value of delay. In effect, the input signal serves as the desired response signal, and for this reason it is always available. In such cases, the linear

adaptive filter attempts to predict future values of the input signal using past samples, giving rise to the name linear prediction for this task.

If an estimate of the signal $x(n+\Delta)$ at time n is desired, a copy of the adaptive filter whose input is the current sample $x(n)$ can be employed to compute this quantity. However, linear prediction has a number of applications besides the obvious application of forecasting future events.

4.3 Interference Cancellation

Interference cancellation refers to situations where it is required to cancel an interfering signal/noise from the given signal which is a mixture of the desired signal and the interference. The principle of interference cancellation is to obtain an estimate of interfering signal and subtract that from the corrupted signal.

Feasibility of this idea relies on the availability of a reference source from which the interfering signal originates. Figure 4.15 depicts the concept of interference cancellation, in its simplest form. There are two inputs to the canceler: *primary* and *reference*. The primary input is the corrupted signal, that is, the desired signal plus interference. The reference input, on the other hand, originates from the interference source only. In some applications of interference cancellation, there might also be some leakage of the desired signal to the reference input. Here, we have ignored this situation for simplicity. The adaptive filter is adjusted so that a replica of the interference signal that is present in the primary signal appears at its output, $y(n)$. Subtracting this from the primary input results in an output which is cleared from interference, thus the name interference cancellation.

We note that the interference cancellation configuration of Figure 4.15 is different from the previous cases of adaptive filters, in the sense that the residual error (which was discarded in other cases) is the cleaned-up signal, here. The desired signal in the previous cases has been replaced here by a noisy (corrupted) version of the actual desired signal.

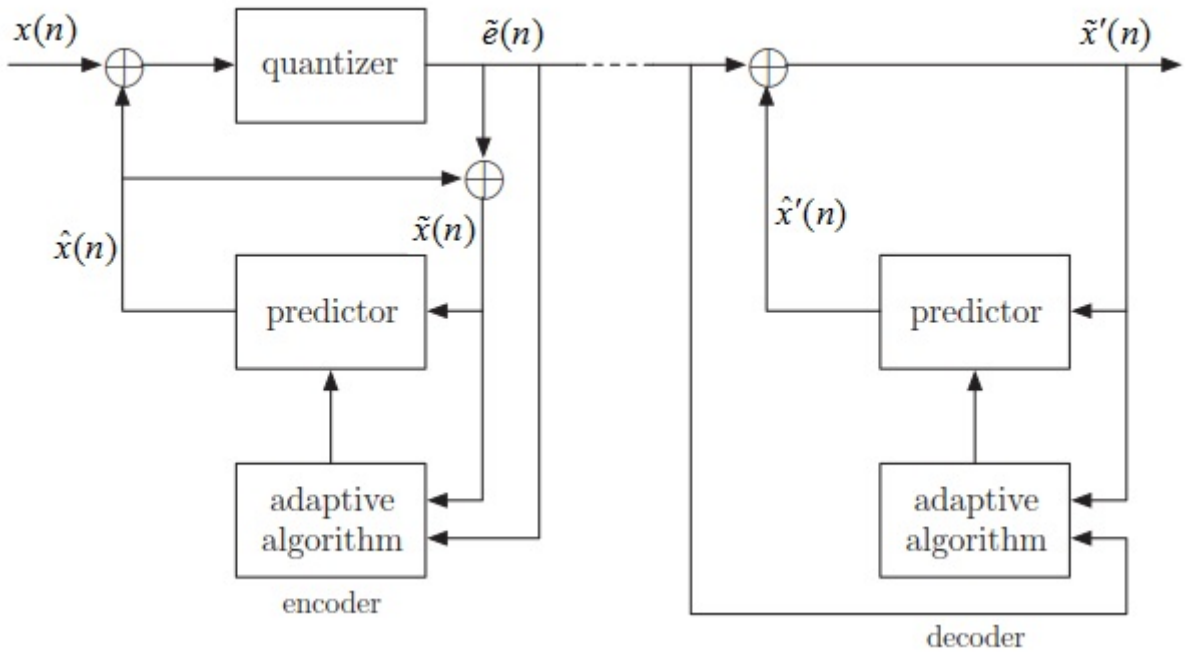


Figure 4.14 – ADPCM encoder–decoder.

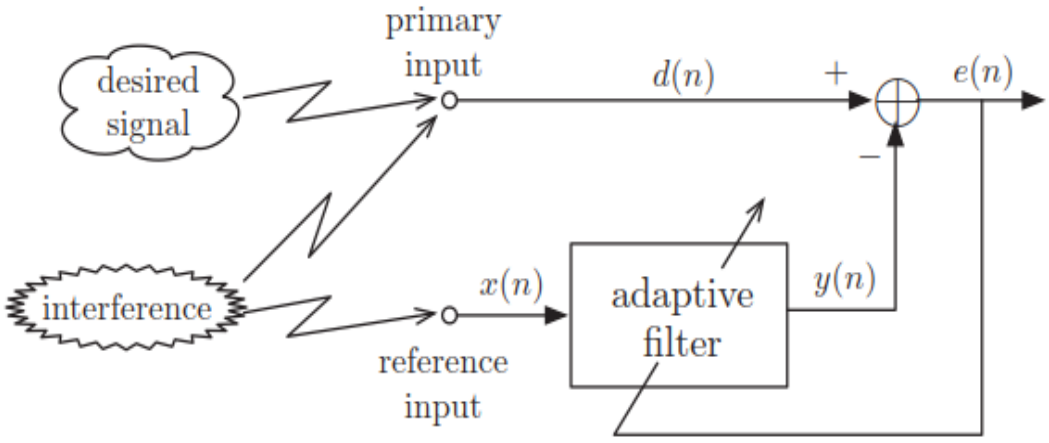


Figure 4.15 – Interference cancellation.

Moreover, the use of the term *reference* to refer to the adaptive filter input is clearly related to the role of this input in the canceler.

In the rest of this section, we present some specific applications of interference canceling.

4.3.1 Echo Cancellation in Telephone Lines

Echoes in telephone lines mostly occur at points where hybrid circuits are used to convert four-wire networks to two-wire ones. Figure 4.16 presents a simplified diagram of a telephone connection network, highlighting the points where echoes occur.

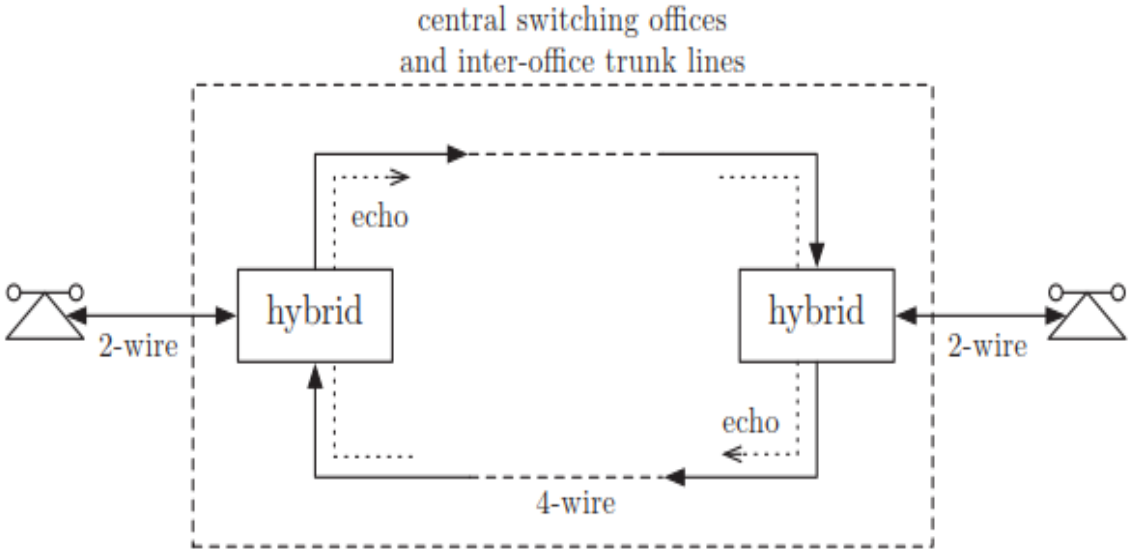


Figure 4.16 – Simplified diagram of a telephone network.

The two wires at the ends are subscriber loops connecting customers' telephones to central offices. It may also include some portions of the local network. The four wires, on the other hand, are carrier systems (trunk lines) for medium-to-long-haul transmission. The distinction is that the two-wire segments carry signals in both directions on the same lines, while in the four-wire segment signals in the two directions are transmitted on two separate lines. Accordingly, the role of hybrid circuit is to separate the signals in the two directions. Perfect operation of the hybrid circuit requires that the incoming signal from the trunk lines should be directed to the subscriber line and that there should not be any leakage (echo) of that to the return line. In practice, however, such ideal behavior cannot be expected from hybrid circuits. There would always be some echo on the return path. In the case of voice

communications (i.e., ordinary conversation on telephone lines), effect of the echoes becomes more obvious (and annoying to the speaker) in long-distance calls, where the delay with which the echo returns to the speaker may be in the range of a few hundred milliseconds. In digital data transmission, both short- and long-delay echoes are serious.

As was noted before and also can clearly be seen from Figure 4.17, the problem of echo cancellation may be viewed as one of system modeling. An adaptive filter is put between the incoming and outgoing lines of the hybrid. By adapting the filter to realize an approximation of the echo path, a replica of the echo is obtained at its output. This is then subtracted from the outgoing signal to clear that from the undesirable echo.

Echo cancelers are usually implemented in transversal form. The time spread of echoes in a typical hybrid circuit is in the range of 20–30 ms. If we assume a sampling rate of 8 kHz for the operation of the echo canceler, an echo spread of 30 ms requires an adaptive filter with at least 240 taps (30 ms×8 kHz). This is a relatively long filter, requiring a high-speed digital signal processor for its realization. Frequency domain processing is often used to reduce the high computational complexity of long filters.

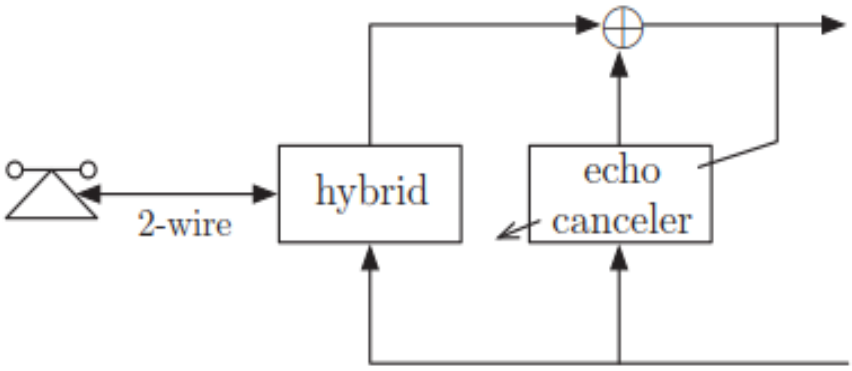


Figure 4.17 – Adaptive echo canceler.

The echo cancelers described previously are applicable to both voice and data transmission. However, more stringent conditions have to be satisfied in the case of data transmission. To maximize the usage of the available bandwidth, full-duplex data transmission is often used. This requires the use of a hybrid circuit for connecting the data modem to the two-wire subscriber loop, as shown in Figure 4.18.

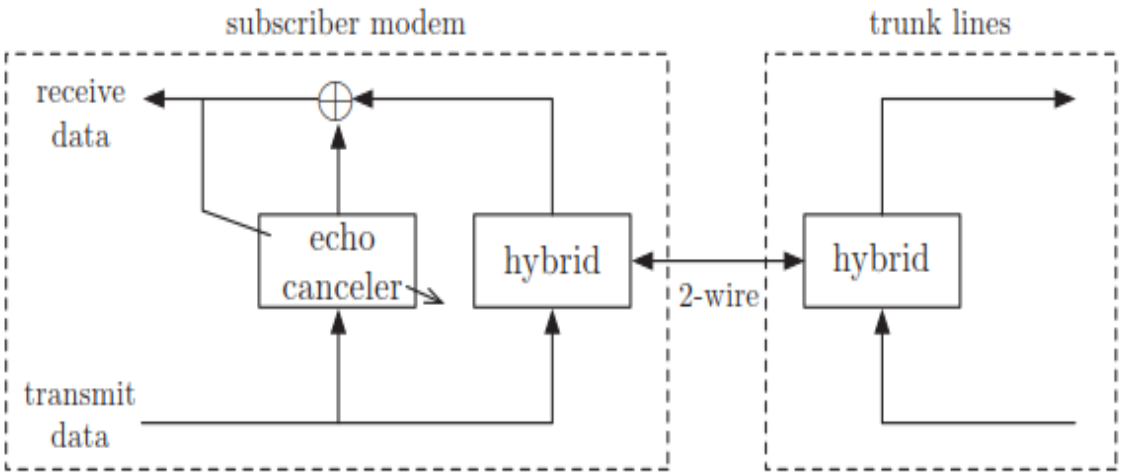


Figure 4.18 – Data echo canceler

The leakage of the transmitted data back to the receiver input is thus inevitable and an echo canceler has to be added, as indicated in Figure 4.18. However, we note that the data echo cancelers are different from the voice echo cancelers used in central switching offices for many ways. For instance, because the input to the data echo canceler are data symbols, it can operate at the data symbol rate that is in the range of 2.4–3 kHz (about three times smaller than the 8 kHz sampling frequency used in voice echo cancelers). For a given echo spread, a lower sampling frequency implies a smaller number of taps for the echo canceler. Clearly, this greatly simplifies the implementation of the echo canceler. On the other hand, the data echo cancelers must achieve a much higher level of echo cancellation to ensure reliable transmission of data at higher bit rates. In addition, one should also take care of echoes returned from the other side of the trunk lines. Detailed discussions on these issues can be found in Lee and Messerschmitt (1994) and Gitlin, Hayes, and Weinstein (1992).

4.3.2 Acoustic Echo Cancellation

The problem of acoustic echo cancellation can be best explained by referring to Figure 4.19, which depicts the scenario that arises in teleconferencing applications.

The speech signal from a far-end speaker, received through a communication channel, is broadcast by a loudspeaker in a room and its echo is picked up by a microphone. This echo must be canceled to prevent its feedback to the far-end speaker. The microphone also picks up the near-end speaker(s) speech and possible background noise, which may exist in the room. An adaptive transversal filter with sufficient length is used to model the acoustics of the room. A replica of the loudspeaker echo is then obtained and subtracted from the microphone signal before the transmission.

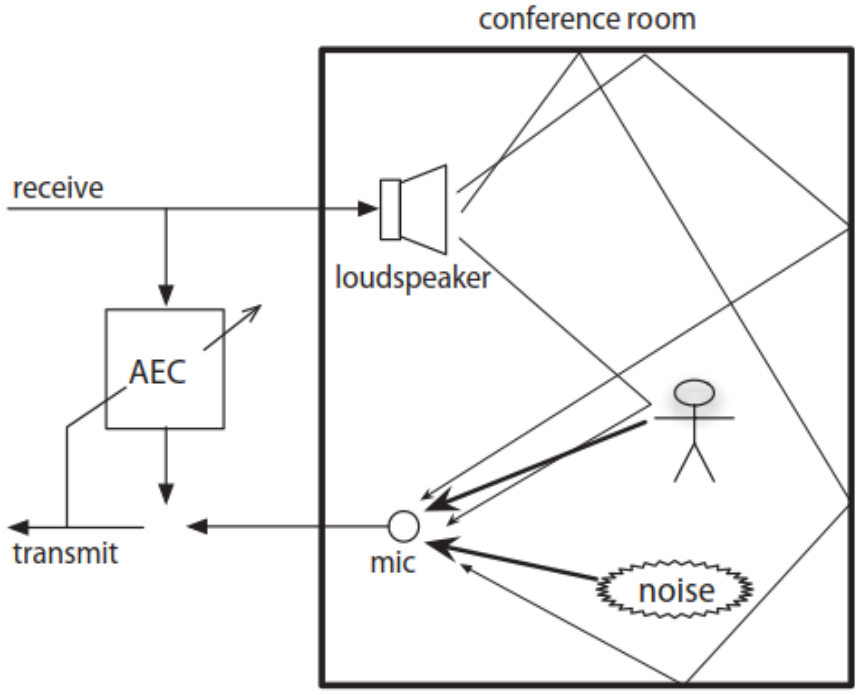


Figure 4.19 – Acoustic echo cancellation

Clearly, the problem of acoustic echo cancellation can also be posed as one of system modeling. The main challenge here is that the echo paths spread over a relatively long length in time. For typical office rooms, echoes in the range of 100–

250 ms spread is quite common. For a sampling rate of 8 kHz, this would mean 800–2000 taps! Thus, the main problem of acoustic echo cancellation is that of realizing very long adaptive filters. In addition, as speech is a lowpass signal, it becomes necessary to use special algorithms to ensure fast adaptation of the echo canceler.

4.4 Channel Equalization

As can be seen from Fig. 4.15, channel equalization or inverse filtering consists of estimating a transfer function to compensate for the linear distortion caused by the channel. From another point of view, the objective is to force a prescribed dynamic behavior for the cascade of the channel (unknown system) and the adaptive filter, determined by the input signal. The first interpretation is more appropriate in communications, where the information is transmitted through dispersive channels. The second interpretation is appropriate for control applications, where the inverse filtering scheme generates control signals to be used in the unknown system.

In the ideal situation, where n(k) = 0 and the equalizer has sufficient order, the error signal is zero if

$$W(z)H(z) = z^{-L} \tag{4.33}$$

where $W(z)$ and $H(z)$ are the equalizer and unknown system transfer functions, respectively. Therefore, the ideal equalizer has the following transfer function

$$W(z) = \frac{z^{-L}}{H(z)}. \tag{4.34}$$

From the above equation, we can conclude that if $H(z)$ is an IIR transfer function with nontrivial numerator and denominator polynomials, $W(z)$ will also be IIR. If $H(z)$ is an all-pole model, $W(z)$ is FIR. If $H(z)$ is an all-zero model, $W(z)$ is an all-pole transfer function.

By applying the inverse Z-transform to equation (4.33), we can conclude that the optimal equalizer impulse response convolved with the channel impulse response

produces an impulse as a result. This means that for zero additional error in the channel, the output signal $y(k)$ restores $x(k-L)$ and, therefore, one can conclude that a deconvolution process took place.

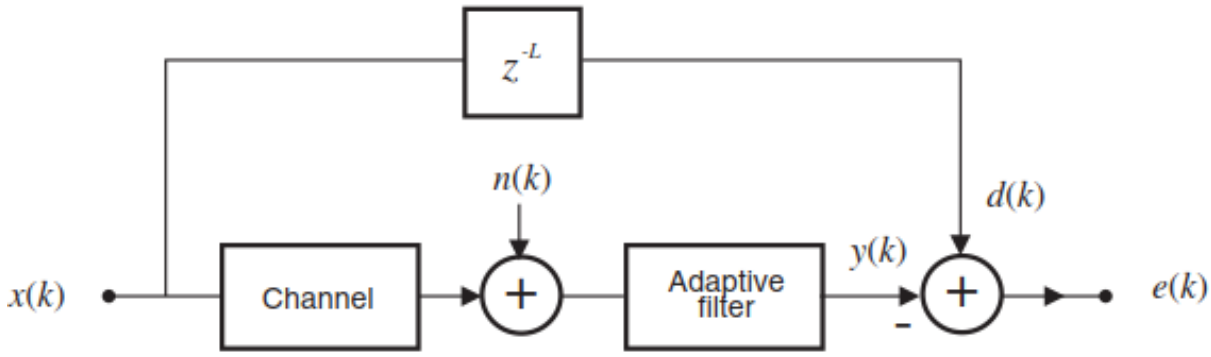


Figure 4.20 – Channel equalization

The delay in the reference signal plays an important role in the equalization process. Without the delay, the desired signal is $x(k)$, whereas the signal $y(k)$ will be mainly influenced by old samples of the input signal, since the unknown system is usually causal. As a consequence, the equalizer should also perform the task of predicting $x(k)$ simultaneously with the main task of equalizing the channel. The introduction of a delay alleviates the prediction task, leaving the equalizer free to invert the channel response. In practice, the reader should try different delays.

In the case the unknown system is not of minimum phase, i.e., its transfer function has zeros outside the unit circle of the Z plane, the optimum equalizer is either stable and noncausal, or unstable and causal. Both solutions are unacceptable. The noncausal stable solution could be better approximated by a causal FIR filter when the delay is included in the desired signal. The delay forces a time shift in the ideal impulse response of the equalizer, allowing the time span, where most of the energy is concentrated, to be in the *causal* region.

If channel noise signal is present and is uncorrelated with the channel's input signal, the error signal and $y(k)$ will be accordingly noisier. However, it should be noticed that the adaptive equalizer, in the process of reducing the MSE, disturbs the

optimal solution by trying to reduce the effects of $n(k)$. Therefore, in a noisy environment the equalizer transfer function is not exactly the inverse of $H(z)$.

In practice, the noblest use of the adaptive equalizer is to compensate for the distortion caused by the transmission channel in a communication system. The main distortions caused by the channels are high attenuation and intersymbol interference (ISI). The ISI is generated when different frequency components of the transmitted signals arrive at different times at the receiver, a phenomenon caused by the nonlinear group delay of the channel. For example, in a digital communication system, the time-dispersive channel extends a transmitted symbol beyond the time interval allotted to it, interfering in the past and future symbols. Under severe ISI, when short symbol space is used, the number of symbols causing ISI is large.

The channel impulse response is a time spread sequence described by $h(k)$ with the received signal being given by

$$re(k+J) = x(k)h(J) + \sum_{l=-\infty, l\neq k}^{k+j} x(l)h(k+J-l) + n(k+J) \qquad (4.35)$$

where J denotes the channel time delay (including the sampler phase). The first term of the above equation corresponds to the desired information, the second term is the interference of the symbols sent before and after $x(k)$. The third term accounts for channel noise. Obviously only the neighboring symbols have significant influence in the second term of the above equation. The elements of the second term involving $x(l)$, for $l > k$, are called pre-cursor ISI since they are caused by components of the data signal that reach the receiver before their cursor. On the other hand, the elements involving $x(l)$, for $l < k$, are called post-cursor ISI.

In many situations, the ISI is reduced by employing an equalizer consisting of an adaptive FIR filter of appropriate length. The adaptive equalizer attempts to cancel the ISI in the presence of noise. In digital communication, a decision device is placed after the equalizer in order to identify the symbol at a given instant. The equalizer coefficients are updated in two distinct circumstances by employing different

reference signals. During the equalizer training period, a previously chosen training signal is transmitted through the channel and a properly delayed version of this signal, that is prestored in the receiver end, is used as reference signal. The training signal is usually a pseudo-noise sequence long enough to allow the equalizer to compensate for the channel distortions. After convergence, the error between the adaptive-filter output and the decision device output is utilized to update the coefficients. The resulting scheme is the decision-directed adaptive equalizer. It should be mentioned that in some applications no training period is available. Usually, in this case, the decision-directed error is used all the time.

A more general equalizer scheme is the decision-feedback equalizer (DFE) illustrated in Fig. 4.16. The DFE is widely used in situations where the channel distortion is severe. The basic idea is to feed back, via a second FIR filter, the decisions made by the decision device that is applied to the equalized signal. The second FIR filter is preceded by a delay, otherwise there is a delay-free loop around the decision device. Assuming the decisions were correct, we are actually feeding back the symbols $x(l)$, for $l < k$, of equation (4.35). The DFE is able to cancel the post-cursor ISI for a number of past symbols (depending on the order of the FIR feedback filter), leaving more freedom for the feedforward section to take care of the remaining terms of the ISI. Some known characteristics of the DFE are:

- The signals that are fed back are symbols, being noise free and allowing computational savings.
- The noise enhancement is reduced, if compared with the feedforward-only equalizer.
- Short time recovery when incorrect decisions are made.
- Reduced sensitivity to sampling phase.

The DFE operation starts with a training period where a known sequence is transmitted through the channel, and the same sequence is used at the receiver as the desired signal. The delay introduced in the training signal is meant to compensate for the delay the transmitted signal faces when passing through the channel. During the

training period the error signal, which consists of the difference between the delayed training signal and signal $y(k)$, is minimized by adapting the coefficients of the forward and feedback filters. After this period, there is no training signal and the desired signal will consist of the decision device output signal. Assuming the decisions are correct, this *blind* way of performing the adaptation is the best solution to keep track of small changes in the channel behavior.

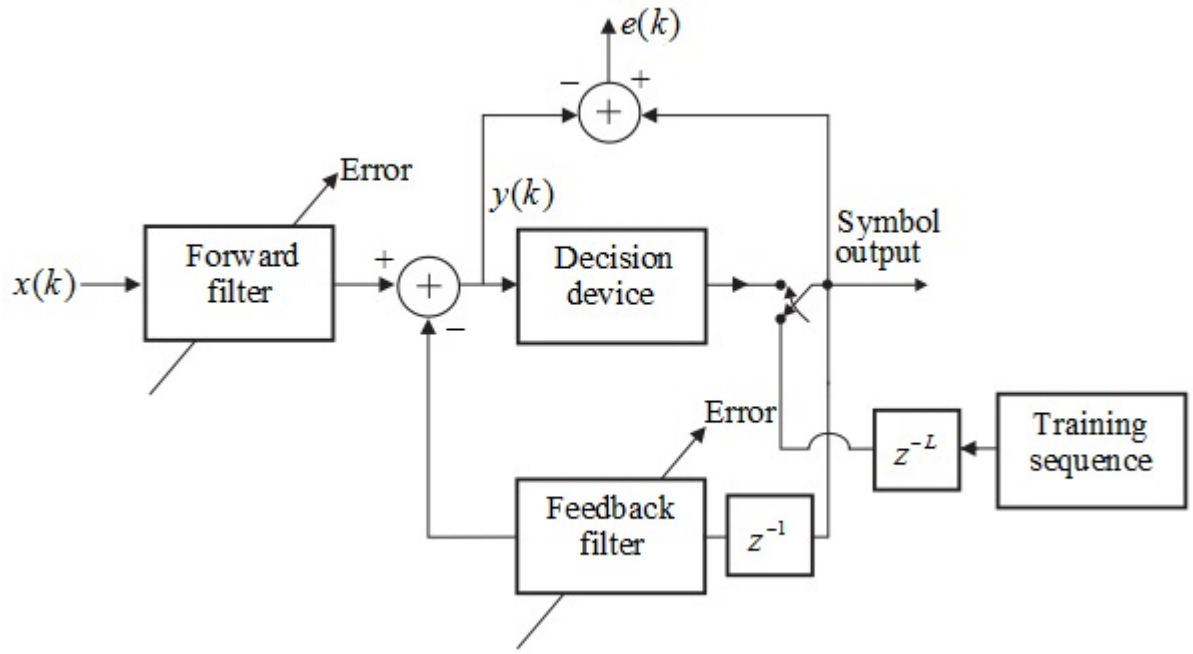


Figure 4.21 – Decision-feedback equalizer

In the examples given below, we will verify the effectiveness of the Wiener solution in environments related to the applications of noise cancellation, prediction, equalization, and identification.

(a) In a noise cancellation environment a sinusoid is corrupted by noise as follows

$$d(k) = \cos(\omega_0 k) + n_1(k)$$

with

$$n_1(k) = -an_1(k-1) + n(k),$$

$|a| < 1$ and $n(k)$ is a zero-mean white noise with variance $\sigma_n^2 = 1$. The input signal of the Wiener filter is described by

$$n_2(k) = -an_2(k-1) + n(k)$$

where $|b| < 1$.

(b) In a prediction case the input signal is modeled as

$$x(k) = -ax(k-1) + n(k),$$

with $n(k)$ being a white noise with unit variance and $|a| < 1$.

(c) In an equalization problem a zero-mean white noise signal $s(k)$ with variance c is transmitted through a channel with an AR model described by

$$\hat{x}(k) = -a\hat{x}(k-1) + s(k)$$

with $|a| < 1$ and the received signal given by

$$x(k) = \hat{x}(k) + n(k)$$

whereas $n(k)$ is a zero-mean white noise with variance d and uncorrelated with $s(k)$.

(d) In a system identification problem a zero-mean white noise signal $x(k)$ with variance c is employed as the input signal to identify an AR system whose model is described by

$$v(k) = -av(k-1) + x(k)$$

where $|a|<1$ and the desired signal is given by

$$d(k) = v(k) + n(k).$$

Repeat the problem if the system to be identified is an MA whose model is described by

$$v(k) = -ax(k-1) + x(k)$$

For all these cases describe the Wiener solution with two coefficients and comment on the results.

Solution:

Some results used in the examples are briefly reviewed. A 2×2 matrix inversion is performed as

$$R^{-1} = \frac{1}{r_{11}r_{22} - r_{12}r_{21}} \begin{pmatrix} r_{22} & -r_{12} \\ -r_{21} & r_{11} \end{pmatrix},$$

where r_{ij} is the element of row i and column j of the matrix R. For two first-order AR modeled signals $x(k)$ and $v(k)$, whose poles are respectively placed at $-a$ and $-b$ with the same white noise input with unit variance, their cross-correlations are given by

$$E\left[x(k)v(k-l)\right] = \frac{(-a)^l}{1-ab}$$

for $l > 0$, and

$$E\left[x(k)v(k-l)\right] = \frac{(-b)^{-l}}{1-ab}$$

for $l < 0$.

(a) The input signal in this case is given by $n_2(k)$, whereas the desired signal is given by $d(k)$. The elements of the correlation matrix are computed as

$$E\left[n_2(k)n_2(k-l)\right] = \frac{(-b)^{|l|}}{1-b^2}.$$

The expression for the cross-correlation vector is given by

$$P=\begin{pmatrix} E\left[\cos\left(\omega_0 k+n_1(k)\right)n_2(k)\right] \\ E\left[\cos\left(\omega_0 k+n_1(k)\right)n_2(k-1)\right]\end{pmatrix}=\begin{pmatrix} E\left[n_1(k)n_2(k)\right] \\ E\left[n_1(k)n_2(k-1)\right]\end{pmatrix}=$$

$$=\begin{pmatrix} \dfrac{1}{1-ab}\sigma_n^2 \\ -\dfrac{a}{1-ab}\sigma_n^2\end{pmatrix}=\begin{pmatrix} \dfrac{1}{1-ab} \\ -\dfrac{a}{1-ab}\end{pmatrix}.$$

The coefficients corresponding to the Wiener solution are given by

$$w_0=R^{-1}p=\begin{pmatrix}1 & b \\ b & 1\end{pmatrix}\begin{pmatrix} \dfrac{1}{1-ab} \\ -\dfrac{a}{1-ab}\end{pmatrix}=\begin{pmatrix} 1 \\ \dfrac{b-a}{1-ab}\end{pmatrix}.$$

The special case where $a = 0$ provides a quite illustrative solution. In this case

$$w_0=\begin{pmatrix}1\\ b\end{pmatrix},$$

so that the error signal is given by

$$e(k)=d(k)-y(k)=\cos(\omega_0 k)+n(k)-w_0^t\begin{pmatrix} n_2(k) \\ n_2(k-1)\end{pmatrix}=$$

$$=\cos(\omega_0 k)+n(k)-n_2(k)-bn_2(k-1)=$$

$$=\cos(\omega_0 k)+n(k)+bn_2(k-1)-n(k)-bn_2(k-1)=\cos(\omega_0 k).$$

As can be observed, the cosine signal is fully recovered since the Wiener filter was able to restore $n(k)$ and remove it from the desired signal.

(b) In the prediction case the input signal is $x(k)$ and the desired signal is $x(k + L)$. Since

$$E\left[x(k)x(k-L)\right]=\frac{(-a)^{|L|}}{1-a^2},$$

the input signal correlation matrix is

$$R=\begin{pmatrix} E\left[x^2(k)\right] & E\left[x(k)x(k-1)\right] \\ E\left[x(k)x(k-1)\right] & E\left[x^2(k-1)\right] \end{pmatrix}=\begin{pmatrix} \dfrac{1}{1-a^2} & -\dfrac{a}{1-a^2} \\ -\dfrac{a}{1-a^2} & \dfrac{1}{1-a^2} \end{pmatrix}.$$

Vector p is described by

$$p=\begin{pmatrix} E\left[x(k+L)x(k)\right] \\ E\left[x(k+L)x(k-1)\right] \end{pmatrix}=\begin{pmatrix} \dfrac{(-a)^{|L|}}{1-a^2} \\ \dfrac{(-a)^{|L+1|}}{1-a^2} \end{pmatrix}.$$

The expression for the optimal coefficient vector is easily derived.

$$w_0=R^{-1}p=$$

$$=\left(1-a^2\right)\begin{pmatrix} \dfrac{1}{1-a^2} & \dfrac{a}{1-a^2} \\ \dfrac{a}{1-a^2} & \dfrac{1}{1-a^2} \end{pmatrix}\begin{pmatrix} \dfrac{(-a)^{L}}{1-a^2} \\ \dfrac{(-a)^{L+1}}{1-a^2} \end{pmatrix}=\begin{pmatrix} (-a)^{L} \\ 0 \end{pmatrix},$$

where in the above equation the value of L is considered positive. The predictor result tells us that an estimate $\hat{x}(k+L)$ of $x(k+L)$ can be obtained as

$$\hat{x}(k+L)=(-a)^{L}x(k).$$

According to our model for the signal $x(k)$, the actual value of $x(k+L)$ is

$$x(k+L)=(-a)^{L}x(k)+\sum_{i=0}^{L-1}(-a)^{i}n(k-i).$$

The results show that if $x(k)$ is an observed data at a given instant of time, the best estimate of $x(k+L)$ in terms of $x(k)$ is to average out the noise as follows

$$\hat{x}(k+L)=(-a)^{L}x(k)+E\left[\sum_{i=0}^{L-1}(-a)^{i}n(k-i)\right]=(-a)^{L}x(k)$$

because $E\left[n(k-i)\right]=0$.

(c) In this equalization problem, matrix R is given by

$$R=\begin{pmatrix} E\left[x^2(k)\right] & E\left[x(k)x(k-1)\right] \\ E\left[x(k)x(k-1)\right] & E\left[x^2(k-1)\right] \end{pmatrix}=\begin{pmatrix} \frac{1}{1-a^2}c+d & -\frac{a}{1-a^2}c \\ -\frac{a}{1-a^2}c & \frac{1}{1-a^2}c+d \end{pmatrix}.$$

By utilizing $s(k-L)$ as desired signal and recalling that it is a white noise and is uncorrelated with the other signals involved in the experiment, the cross-correlation vector between the input and desired signals is

$$p=\begin{pmatrix} E\left[x(k)s(k-L)\right] \\ E\left[x(k-1)s(k-L)\right] \end{pmatrix}=\begin{pmatrix} (-1)^L a^L c \\ (-1)^{L-1} a^{L-1} c \end{pmatrix}.$$

The coefficients of the underlying Wiener solution are given by

$$w_0 = R^{-1}p =$$

$$=\frac{1}{\frac{c^2}{1-a^2}+2\frac{dc}{1-a^2}+d^2}\begin{pmatrix} \frac{1}{1-a^2}c+d & \frac{a}{1-a^2}c \\ \frac{a}{1-a^2}c & \frac{1}{1-a^2}c+d \end{pmatrix}\begin{pmatrix} (-1)^L a^L c \\ (-1)^{L-1} a^{L-1} c \end{pmatrix}=$$

$$=\frac{(-1)^L a^L c}{\frac{c^2}{1-a^2}+2\frac{cd}{1-a^2}+d^2}\begin{pmatrix} \frac{c}{1-a^2}+d-\frac{c}{1-a^2} \\ \frac{ac}{1-a^2}-a^{-1}d-\frac{a^{-1}c}{1-a^2} \end{pmatrix}=$$

$$=\frac{(-1)^L a^L c}{\frac{c^2}{1-a^2}+2\frac{cd}{1-a^2}+d^2}\begin{pmatrix} d \\ -a^{-1}d-a^{-1}c \end{pmatrix}.$$

If there is no additional noise, i.e. $d = 0$, the above result becomes

$$w_0 = \begin{pmatrix} 0 \\ (-1)^{L-1}\left(1-a^2\right)a^{L-1} \end{pmatrix}.$$

That is, the Wiener solution is just correcting the gain of the previously received component of the input signal, namely $x(k-1)$, while not using its most recent component $x(k)$. This happens because the desired signal $s(k-L)$ at instant k has a defined correlation with any previously received symbol. On the other hand, if the signal $s(k)$ is a colored noise, the Wiener filter would have a nonzero first coefficient in a noiseless environment. In case there is environmental noise, the solution tries to find a perfect balance between the desired signal modeling and the noise amplification.

(d) In the system identification example the input signal correlation matrix is given by

$$R = \begin{pmatrix} c & 0 \\ 0 & c \end{pmatrix}.$$

With the desired signal $d(k)$, the cross-correlation vector is described as

$$p = \begin{pmatrix} E\lfloor x(k)d(k) \rfloor \\ E\lceil x(k-1)d(k) \rceil \end{pmatrix} = \begin{pmatrix} c \\ -ca \end{pmatrix}.$$

The coefficients of the underlying Wiener solution are given by

$$w_0 = R^{-1}p = \begin{pmatrix} \frac{1}{c} & 0 \\ 0 & \frac{1}{c} \end{pmatrix} \begin{pmatrix} c \\ -ca \end{pmatrix} = \begin{pmatrix} 1 \\ -a \end{pmatrix}.$$

Note that this solution represents the best way in which a first-order FIR model can approximate an IIR model because

$$W_0(z) = 1 - az^{-1}$$

and

$$\frac{1}{1+az^{-1}} = 1 - az^{-1} + a^2z^{-2} + \ldots$$

On the other hand, if the unknown model is the described FIR model such as $v(k)=-ax(k-1)+x(k)$, the Wiener solution remains the same and corresponds exactly to the unknown system model.

In all these examples, the environmental signals are considered wide-sense stationary and their statistics is assumed to be known. In a practical situation, not only the statistics might be unknown, but the environments are usually nonstationary as well. In these situations, the adaptive filters come into play because their coefficients vary with time according to measured signals from the environment.

References

1 Cowan C.F.N., Grant P.M., Adams P.F. Adaptive filters. – Upper Saddle River, New Jersey: Prentice-Hall, 1985. – 308 p.

2 B.Widrow and S.D.Stearns, Adaptive Signal Processing. – Upper Saddle River, New Jersey: Prentice-Hall, 1985. – 491 p.

3 Honig M. L., Messerschmitt D. G. Adaptive filters: structures, algorithms and applications. – MA, Hingham: Kluwer Academic Publishers, 1984. – 337 p.

4 Giordano A. A., Hsu F. M. Least square estimation with application to digital signal processing. – Canada, Toronto: John Wiley and Sons, Inc., 1985. – 412 p.

5 Alexander S. T. Adaptive signal processing. Theory and applications. – New York: Springer Verlag, 1986. – 179 p.

6 Ljung L., Soderstrom T. Theory and practice of recursive identification. – Cambridge, Massachusetts, London, England: MIT Press, 1986. – 529 p.

7 Treicher J. R., Johnson C. R., Larimore M. G. Theory and design of adaptive filters. – New York: John Wiley and Sons, Inc., 1987. – 342 p.

8 Orfanidis S. J. Optimum signal processing. – New York: McGraw-Hill, Inc., 1988. – 590 p.

9 Kalouptsidis N., Theodoridis S., Eds. Adaptive system identification and signal processing algorithms. – NJ, Englewood Cliffs: Prentice-Hall, 1993. – 560 p.

10 Clarkson P. M. Optimum and adaptive signal processing. – London: CRC Press, 1993. – 529 p.

11 Zelniker G., Taylor F. J. Advanced digital signal processing: theory and applications. – New York: Marcel Dekker, Inc., 1994. – 666 p.

12 Regalia P. A. Adaptive IIR filtering in signal processing and control. – New York: Marcel Dekker, Inc., 1995. – 678 p.

13 Macchi O. Adaptive processing. The least mean squares approach with application in transmission. – New York: John Wiley and Sons. – 1995. – 456 p.

14 Farhang-Boroujeny B. Adaptive filters theory and applications. – New York: John Willey & Sons, 1999. – 548 p.

15 Bellanger M. G., Adaptive digital filters. Second edition. – New York, Marcel Dekker, Inc., 2001. – 451 p.

16 Haykin S. Adaptive filter theory (4-th edition). – Prentice Hall, 2001. – 936 p.

17 Sayed A. H. Fundamentals of adaptive filtering. – NJ, Hoboken: John Wiley and Sons, Inc., 2003. – 1125 p.

18 Ratynskii M. V. Adaptation and superresolution in antenna arrays. – Moscow: Radio i Sviaz, 2003 – 200p. (in Russian)

19 Benesty J., Huang Y. (Eds). Adaptive signal processing: applications to realworld problems. – Berlin, Heidelberg, New York, Springer-Verlag, 2003. – 356 p.

20 Manolakis D. G., Ingle V. K., Kogon S. M. Statistical and adaptive signal processing. – Boston: Artech House, 2005. – 796 p.

21 Tarakanov A. N., Khriashchev V. V., Priorov A. L. Adaptive digital signal processing: tutorial. – Yaroslavl: Demidov Yaroslavl State Unoversity, 2001. – 134 p. (in Russian)

22 Sergienko A. B. Digital signal processing: tutorial. Third edition – Saint-Petersburg: BHV, 2011. – 768 p. (in Russian)

23 Ogunfunmi T. Adaptive nonlinear system identification: the Volterra and Wiener model approaches. – Springer Science + Business Media, LCC. – 2007. – 230 p.

24 Diniz P. S. R. Adaptive filtering algorithms and practical implementation. Third edition. – New York, Springer Science + Business Media, 2008. – 627 p.

25 Sayed A. H. Adaptive filters. – NJ, Hoboken: John Wiley and Sons, Inc., 2008. – 785 p.

26 Mandic D. P., Goh V. S. L. Complex valued nonlinear adaptive filters. – UK,

Chichester, 2009. – 324 p.

27 Lee K.-A., Gan W.-S., Kuo S. M. Subband Adaptive Filtering: Theory and Implementation. – UK, West Sussex: John Wiley and Sons, Ltd., 2009. – 324 p.

28 Apolinario J. A., Ed. QRD-RLS Adaptive Filtering. – Springer, 2009. – 356 p.

29 Dzhigan V. I. Adaptive filters and their application in radiotechnics and communication. Part 1. // Modern Electronics – 2009. – No. 9. – P. 56–63. (in Russian)

30 Dzhigan V. I. Adaptive filters and their application in radiotechnics and communication. Part 2. // Modern Electronics – 2010. – No. 1. – P. 72–77. (in Russian)

31 Dzhigan V. I. Adaptive filters and their application in radiotechnics and communication. Part 3. // Modern Electronics – 2010. – No. 2. – P. 70–77. (in Russian)

32 Makino S. Acoustic echo cancellation // IEEE Signal Processing Magazine. – 1997. – Vol. 14. – №5. – P. 39–41.

33 Messerschmitt D. Echo cancellation in speech and data transmission // IEEE Journal on Selected Areas in Communications. – 1984. – Vol. 2. – №2. – P. 283–297.

34 Qureshi S. Adaptive equalization // IEEE Communications Magazine. – 1982. – Vol. 20. – №2. – P. 9–16.

35 Allen J. B., Berkley D. A. Image method for efficiently simulating small-room acoustics // The Journal Acoustic Society of America. – 1979. – Vol. 65. – №4. – P. 943–950.

36 Digital network echo cancellers // ITU-T Recommendation G.168. Series G: Transmission systems and media, digital systems and networks. International telephone connections and circuits – Apparatus associated with long-distance telephone circuits. 04/2000. – Geneva. – 2001. – 116 p.

37 Deller J. R., Proakis J. G., Hansen G. H. L. Discrete-time processing of speech signals. – Upper Saddle River, NJ: Prentice Hall, 1993. – 908 p.

38 Chen Y., Le-Ngoc T., Champagne B., Xu C. Recursive least squares constant modulus algorithm for blind adaptive array // IEEE Trans. Signal Processing. – 2004. – Vol. 52. – №5. – P. 1452–1456.

39 Widrow B., McCool J., Ball M. The complex LMS algorithm // Proceedings of the IEEE. – 1975. – Vol. 63. – №4 – P. 719–720.

40 Frost O. L. An algorithm for linearly constrained adaptive array processing // Proceedings of the IEEE. – 1972. – Vol. 60. – №8. – P. 926–935.

41 Su Y.-L. A complex algorithm for linearly constrained adaptive arrays // IEEE Antennas and Propagation. – 1983. – Vol. 31. – №4. – P. 676–678.

42 Reed I. S. Brief history of adaptive arrays // Proceedings of the Military Communication Conference. – 1985. – Vol. 2,3. – P. 515–518.

43 Widrow B., Mantey P. E., Griffiths L. J., Goode B. B. Adaptive antenna systems // Proceedings of the IEEE. – 1967. – Vol. 55. – №12. – P. 2143–2159.

44 Becker F. K., Holzman L. N., Lucky R. W., Port E. Automatic equalization for digital communication // Proceedings of the IEEE. – 1965. – Vol. 52. – №1. – P. 96–97.

45 Lucky R. W. The adaptive equalizer // IEEE Signal Processing Magazine. – 2006. – Vol. 23. – №3. – P. 104–107.

46 Dentino M., McCool J., Widrow B. Adaptive filtering in frequency domain // Proceedings of the IEEE. – 1978. – Vol. 66. – №12. – P. 1658–1659.

47 Shynk J. J. Frequency-domain and multirate adaptive filtering // IEEE Signal Processing Magazine. – 1992. – Vol. 9. – №1. – P. 14–37.

48 Brandwood D. H. A complex gradient operator and its applications in adaptive array theory // IEE Proceedings. Parts F and H. – 1983. – Vol. 130. – №1. – P. 11–16.

49 Voevodin V. V., Kuznetsov Yu. A. Matrices and Calculations. – Moscow: Nauka. – 1984. – 320 p. (in Russian)

50 Belousov I. V. Matrices and determinants. – Chisinau: Institute of Applied Physics of Moldavian Academy of Sciences, 2006. – 101 p. (in Russian)

51 Widrow B. Adaptive filters I: fundamentals. – Stanford Electronics Laboratories Technical Report 6764-6. – December 1966. – 58 p.

52 Widrow B., McCool J. M., Larimore M. G., Johnson C. R. Stationary and nonstationary characteristics of the LMS adaptive filter // Proceedings of the IEEE. – 1976. – Vol. 64. – №8. – P. 1151–1162.

53 Nagomo J. I., Noda A. A. learning method for system identification // IEEE Trans. Automatic Control. – 1967. – Vol. 12. – №3. – P. 282–287.

54 Hsia T. C. Convergence analysis of LMS and NLMS adaptive algorithms. // Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing. 983. P. 667–670.

55 Mathews V. J., Xie Z. A stochastic gradient adaptive filter with gradient adaptive step size // IEEE Trans. Signal Processing. – 1993. – Vol. 41. – №6. – P. 2075–2087.

56 Tikhonov V. I. Statistical Radiotechnics. Second edition. – Moscow: Radio i Sviaz, 1982. – 624 p. (in Russian)

57 Golub G. H., C. F. Van Loan. Matrix Computations. Fourth Edition. – Baltimore: Johns Hopkins University Press, 2013. – 780 p.

58 Semushin I. V. Numerical methods: tutorial for universities. – Ulyanovsk: Ulyanovsk State TechnicalUniversity. – 2006. – 178 p. (in Russian)

59 Zinchuk V. M., Sosulin Yu. G., Limarev A. Ye., Mukhin N. P. Adaptive digital filtration of noise-shaped signals in radiotechnical systems // Digital Signal Processing. – 2000. – No 1. – P. 5–18. (in Russian)

60 Sergienko A. B. Adaptive filtration alrogithms: spesifics of realization in MATLAB // Exponenta Pro: mathematics in applications. – 2003. – No. 1. – P. 18–28. (in Russian)

61 Lewis P. S. QR-based algorithms for multichannel adaptive least squares lattice filters // IEEE Trans. Acoustics, Speech and Signal Processing. – 1990. – Vol. 38. – №3. – P. 421–432.

62 Dzhigan V. I. Algorithms of adaptive filtrations of non-stationary signals // D. Sci. Thesis. – Moscow, Scientific Production Center "Electronic computing and information systems", 2006. – 342 p. (in Russian)

63 Solonina A. I., Arbuzov S. M. Digital signal processing. Modeling in MATLAB: tutorial. – Saint-Petersburg: BHV, 2008. – 816 p. (in Russian)

64 Solonina A. I., Digital signal processing. Modeling in Simulink: tutorial. – Saint-Petersburg: BHV, 2008. – 432 p. (in Russian)

65 Dzhigan V. I., Pletneva I. D. Application of CM-algorithm of affine projections with linear restrictions for signal adaptive filtration in an antenna array // Antennas. – 2008. – No. 10. – P. 14–24. (in Russian)

66 Honig M. L., Messerschmitt D. G. Adaptive filters: structures, algorithms and applications. – MA, Hingham: Kluwer Academic Publishers, 1984. – 337 p.

67 Diniz P. S. R., Adaptive Filtering: Algorithms and Practical Implementation. 3rd edition Springer, New York, NY, USA. 2008. – 621 p.

68 Farhang-Boroujeny, B. Adaptive filters: theory and applications / Behrouz Farhang-Boroujeny. – Second edition. – John Wiley & Sons Ltd, United Kingdom. 2013. – 778 p.

69 Qureshi S. U., Adaptive Equalization // Proceedings of the IEEE. – vol. 73. – 1985. – pp. 1349-1387.

70 Lin D. W., Minimum mean-squared error echo cancellation and equalization for digital subscriber line transmission: Part I - theory and computation // IEEE Trans. on Communications. – vol. 38. – 1990. – pp. 31–38.

71 Abdulrahman M., Falconer D. D.. Cyclostationary crosstalk suppression by decision feedback equalization on digital subscriber line. // IEEE Journal on Selected Areas in Communications. – vol. 10. – 1992. – pp. 640-649.

72 Miller S., Childers D., Probability and Random Processes With Applications to Signal Processing and Communications. Second edition. – Amsterdam: Elseiver/Academic Press. – 2012. – 598 p.

73 Solonina A. I., Ulakhovich D. A., Arbuzov S. M., Soloveva Ye. B., Fundamentals of digital signal processing. Second Edition. – Saint-Petersburg: BHV. – 2005 – 768 p. (in Russian)

74 Davenport W. B. and Root W. L., An Introduction to the Theory of Random Signals and Noise. – Wiley-IEEE Press. – 1987. – 407 p.

75 Kleehammer M., Mathematical Development of the Elliptic Filter [Electronic resource]. – Queen's University at Kingston. – 2013. – 71 p., online: https://qspace.library.queensu.ca/bitstream/handle/1974/8209/Filter%20Theory.pdf?sequence=1

76 Lutovac M. D. and Tosic D. V., Elliptic Rational Functions // The Mathematica Journal. – Vol. 9. – 2005. – p. 598-608.

77 Theory of automatic control. Part 2: Theory of non-lineal and special systems of automatic control. – Edited by A. A. Voronov. – Moscow: Vysshaia Shkola. – 1986. – 504 p. (in Russian)

78 Brammer K., Siffling G. Kalman–Bucy filters. – Artech Print on Demand. – 1989. – 404 p.

79 V. S. Gutnikov. Integrated electronics in measuring devices. – Saint-Petersburg: Energoatomizdat. – 1988. – 304 p. (in Russian)

80 Khudaverdyan D. Kalman filter [Electronic resource]. – 2013. – access mode: https://david.wf/kalmanfilter/

Навчальне видання

Гусєв Олександр Юрійович
Горєв В’ячеслав Миколайович
Корнієнко Валерій Іванович

ТЕОРІЯ АДАПТИВНОЇ ФІЛЬТРАЦІЇ

Навчальний посібник

(англійською мовою)

Видано в редакції авторів

Підписано до друку 10.09.2019. Формат 30х42/4.
Папір офсет. Ризографія. Ум. друк. арк. 9,1.
Обл.-вид. арк. 11,6. Тираж 50 пр. Зам. №

Підготовлено до друку та видрукувано в
Національному технічному університеті «Дніпровська політехніка»
Свідоцтво про внесення до Державного реєстру ДК № 1842 від 11.06.2004 р.
49005, м. Дніпро, просп. Д. Яворницького 19.