Robust Training of ADALINA Based on the Criterion of the Maximum Correntropy in the Presence of Outliers and Correlated Noise

Oleg Rudenko and Oleksandr Bezsonov

Kharkiv National University of Radio, Nauky Ave. 14, Kharkiv, 61166, Ukraine

Abstract

In the given paper the main relations that describe an adaptive multi-step algorithm for training ADALINA are obtained. The use of such an algorithm accelerates the learning process by using information not only about one last cycle, but also about a number of previous cycles. The robustness of the estimates is ensured by the application of the maximum correlation criterion.

Keywords 1

ADALINA, optimization, neural network, algorithm, gradient, training, estimation

1. Introduction

ADALINA (Adaptive Linear Element) was the first linear neural network proposed by Widrow B. and Hoff M.E. and represented an alternative to the perceptron [1]. Subsequently, this element and the algorithm for its training found a fairly wide application in problems of identification, control, filtering, etc. The Widrow-Hoff learning algorithm is a Kachmazh algorithm for solving systems of linear algebraic equations. Properties of this algorithm for the solution of the identification problem are described in sufficient detail in [2]. In [3], the Kachmazh (Widrow-Hoff) regularized algorithm was used to train ADALINA in the problem of estimating non-stationary parameters. In this paper, a multistep learning algorithm is considered, which is a recurrent current regression analysis (TPA) algorithm that accelerates the ADALINA learning process by using information not only about one last cycle (as in the Widrow-Hoff algorithm), but also about a number of previous cycles.

2. The task of the ADALINA training

ADALINA is described by the equation

$$y_{n+1} = c^{*T} x_{n+1} + \xi_{n+1}, \tag{1}$$

where y_{n+1} – observed output signal; $x_{n+1} = (x_{1,n+1}, x_{2,n+1}, .. x_{N,n+1})^T$ – vector of the input signals $N \times 1$; $c^* = (c_1^*, c_2^*, .. c_N^*)^T$ – is the vector of the required parameters $N \times 1$; ξ_{n+1} – noise; n – discrete time.

The task of its training is to determine (estimate) the vector of parameters c^* and is reduced to minimizing some preselected quality functional (identification criterion)

$$F[e_n] = \sum_{i=1}^{n} \rho(e_i), \tag{2}$$

COLINS-2021: 5th International Conference on Computational Linguistics and Intelligent Systems, April 22–23, 2021, Kharkiv, Ukraine
EMAIL: oleh.rudenko@nure.ua (O. Rudenko); oleksandr.bezsonov@nure.ua (O. Bezsonov)
ORCID: 0000-0003-0859-2015 (O. Rudenko); 0000-0001-6104-4275 (O. Bezsonov)

where $e_i = y_i - \hat{y}_i$; $\hat{y}_i = c_{i-1}^T x_i$ – output signal of the model; c – vector estimate c^*; $\rho(e_i)$ – some differentiable loss function satisfying the conditions

1) $\rho(e_i) \geq 0$;
2) $\rho(0) = 0$;
3) $\rho(e_i) = \rho(-e_i)$;
4) $\rho(e_i) \geq \rho(e_j)$ $\quad for \quad$ $|e_i| \geq |e_j|$.

The identification task is to find an estimate $\hat{\theta}$ defined as a solution to the extreme minimum problem

$$F(\theta) = \min, \tag{3}$$

or as a solution to the system of equations

$$\frac{\partial F(e)}{\partial \theta_j} = \sum_{i=1}^{n} \rho'(e_i) \frac{\partial e_i}{\partial \theta_j} = 0, \tag{4}$$

where $\rho'(e_i)) = \frac{\partial \rho(e_i)}{\partial e_i}$ – function of influence.

If we introduce the weight function $\omega(e) = \rho'(e)/e$, then the system of equations (4) can be written as follows:

$$\sum_{i=1}^{n} \omega(e_i) e_i \frac{\partial e_i}{\partial \theta_j} = 0, \tag{5}$$

and minimization of functional (2) will be equivalent to minimization of the weighted quadratic functional, which is most often encountered in practice

$$\min \sum_{i=1}^{n} \omega(e_i) e_i^2 \ ... \tag{6}$$

When choosing $\rho(e_i) = 0{,}5e_i^2$ influence function $\rho'(e_i) = e_i$, i.e. grows linearly with increasing e_i, which explains the instability of the LMS estimate to outliers and to interference, the distributions of which have long “tails”.

A robust M-score represents a score c, defined as a solution to the extremal problem (3) or as a solution to the system of equations (4), but the loss function $\rho(e_i)$ should be chosen other than quadratic.

There is a fairly large number of functionals that provide robust M-estimates; however, the most common are the combined functionals proposed by Huber [4] and Hempel [5] and consisting of a quadratic one, which ensures the optimality of estimates for a Gaussian distribution, and a modular one, which makes it possible to obtain a more robust distribution with heavy tails estimate. However, the efficiency of the obtained robust estimates substantially depends on the numerous parameters used in these criteria and selected on the basis of the researcher's experience.

Recently, when solving problems of identification, filtration, etc. robust algorithms that are obtained not on the basis of minimization (3), but on the basis of maximizing the correlation criterion [6–13] are gaining popularity. These algorithms are simple to implement and efficient.

3. Correntropy and algorithms for its maximization

Correntropy, defined as a localized measure of similarity, has proven to be very effective for obtaining robust estimates due to the fact that it is less sensitive to outliers [6–13].

For two random variables X and Y, the correlation is defined as

$$V(X,Y) = M\{k_\sigma(X,Y)\}, \tag{7}$$

where $k_\sigma(\bullet)$ – rotation invariant Mercer kernels; σ – kernel width.

The most widely used in calculating the correlation is Gaussian function, defined by the formula

$$k_\sigma(x,y)=\frac{1}{\sqrt{2\pi}\sigma}\exp\left(-\frac{(x-y)^2}{2\sigma^2}\right). \quad (8)$$

When calculating the correlation, it is necessary to know the joint distribution of random variables X and Y, which are usually unknown. In practice, there is often a finite number of samples $\{x_i, y_i\}, i=1,2,...,N.$ Therefore, the most simple estimate of the correlation is calculated as follows:

$$\hat{V}(X,Y)=\frac{1}{N}\sum_{i=1}^{N}k_\sigma(x_i-y_i). \quad (9)$$

In tasks of identification, filtering, etc. as a functional, the correlation between the required output signal d_i and the output signal of the model y_i ... is used. In case of using Gaussian kernels, the optimized functional takes the form

$$J_{corr}(n)=\frac{1}{\sqrt{2\pi}\sigma}\frac{1}{N}\sum_{i=n-N+1}^{N}\exp\left(-\frac{e_i^2}{2\sigma^2}\right), \quad (10)$$

where $e_i=d_i-y_i$ – identification (filtering) error.

Gradient optimization algorithm (10) with $N=1$ looks like [6–9]

$$w_{n+1}=w_n+\gamma\exp\left(-\frac{e_{n+1}^2}{2\sigma^2}\right)e_n x_{n+1}, \quad (11)$$

where γ – a parameter that affects the convergence rate.

In [12], to eliminate impulse noise, a recurrent weighted least squares method (RWLS) was proposed, which minimizes the criterion

$$\psi_{n+1}=\exp\left(-\frac{e_{n+1}^2}{2\sigma^2}\right) \quad (12)$$

and having the form

$$c_{n+1}=c_n+\frac{\psi_{n+1}P_n x_{n+1}}{\lambda+\psi_{n+1}x_{n+1}^T P_n x_{n+1}}(y_{n+1}-c_n^T x_{n+1}) \quad (13)$$

$$P_{n+1}=\lambda^{-1}\left(P_n-\frac{\psi_{n+1}P_n x_{n+1}x_{n+1}^T P_n}{\lambda+\psi_{n+1}x_{n+1}^T P_n x_{n+1}}\right). \quad (14)$$

Here $0\le\lambda<1-$ weighing coefficient.

Thus, when obtaining the formula for calculating P_{n+1} (14) the approximation

$$P_{n+1}=\lambda P_n+\psi_{n+1}x_{n+1}x_{n+1}^T \quad (15)$$

is used.

As it is known, the introduction into the algorithm of the parameter λ is advisable for identifying non-stationary parameters.

Another approach to estimate nonstationary parameters is to use a limited number of measurements in RLS, which leads to the algorithm of the current regression analysis method [14].

4. Recurrent TPA algorithm with correlated interference

Consider the problem of training ADALINA described by equation (1), which in matrix form (after obtaining information on $n+1-$ iteration) is written like this

$$Y_{n+1}=X_{n+1}c^*+\Xi_{n+1}, \quad (16)$$

where $Y_{n+1}=(y_1,y_2,...y_{n+1})^T$ – vector of output signals;

$X_{n+1}^T=(x_1,x_2,...,x_{n+1})^T$ – matrix of input signals;

$c^* = (c_1^*, c_2^*, ..c_N^*)^T$ – vector of estimated parameters;

$\Xi_{n+1} = (\xi_1, \xi_2, ..., \xi_{n+1})^T$ – is the vector of noise.

Covariance matrix D_n order n interference ξ_{n+1} has the following form

$$D_{n+1} = M\{\Xi_{n+1}\Xi_{n+1}^T\} = \begin{bmatrix} d_{1,1} & d_{1,2}... & d_{1,n} & d_{1,n+1} \\ d_{2,1} & d_{2,2}... & d_{2,n} & d_{2,n+1} \\ ... & ... & ... & ... \\ d_{n+1,1} & d_{n+1,2...} & d_{n+1,n} & d_{n+1,n+1} \end{bmatrix} = \begin{bmatrix} D_n & d_n \\ d_n^T & d_{n+1,n+1} \end{bmatrix},$$

where $d_{ij} = M\{\xi_i \xi_j\}$; $d_n^T = (d_{n,1}, d_{n,2}, ..., d_{n,n}) = M\{\xi_{n+1}\Xi_n^T\}$

As known, the application of the assessment

$$c_{n+1} = \left(X_{n+1}^T X_{n+1}\right)^{-1} X_{n+1}^T Y_{n+1}$$

to the model with correlated noise gives estimates, the variances of which will be underestimated.

The Gaussian-Markov estimate (LMS) obtained by minimizing a quadratic functional has the form

$$c_n = \left(X_{n+1}^T D_{n+1}^{-1} X_{n+1}\right)^{-1} X_{n+1}^T D_{n+1}^{-1} Y_{n+1}. \quad (17)$$

The current regression analysis algorithm, which has the form

$$c_{n+1|L} = (X_{n+1|L}^T X_{n+1|L})^{-1} X_{n+1|L}^T Y_{n+1|L}, \quad (18)$$

where

$$Y_{n+1|L} = \begin{pmatrix} Y_{n|L-1} \\ --- \\ y_{n+1} \end{pmatrix} = \begin{pmatrix} y_{n-L+1} \\ --- \\ Y_{n+1|L-1} \end{pmatrix} - \text{vector } L \times 1; \quad (19)$$

$$X_{n+1|L} = \begin{pmatrix} X_{n|L-1} \\ --- \\ x_{n+1}^T \end{pmatrix} = \begin{pmatrix} x_{n-L+1}^T \\ --- \\ X_{n+1|L-1} \end{pmatrix} - \text{the matrix } (L \times 1) \times N; \quad (20)$$

was proposed in [14]. In [15] a modification of this algorithm is considered, using the mechanism of forgetting the past information (smoothing). Here $L = const (L \geq N)$ – algorithm's memory.

By analogy with the Gaussian-Markov estimate (17), the following estimate can be obtained:

$$c_{n+1|L} = (X_{n+1|L}^T D_{n+1|L}^{-1} X_{n+1|L})^{-1} X_{n+1|L}^T D_{n+1|L}^{-1} Y_{n+1|L}, \quad (21)$$

where

$$D_{n+1|L} = \begin{bmatrix} d_{n-L+1,n-L+1} & d_{n-L,n-L+2}... & d_{n-L+1,n-1} & d_{n-L+1,n+1} \\ d_{n-L+2,n-L+1} & d_{n-L+2,n-L+2} & d_{n-L+2,n-1} & d_{n-L+2,n+1} \\ ... & ... & ... & ... \\ d_{n,n-L+1} & d_{n,n-L+2} & d_{n,n-1} & d_{n+1,n+1} \end{bmatrix} = \begin{bmatrix} D_{n|L-1+1} & \vdots & d_n \\ - & - & - \\ d_n^T & \vdots & d_{n+1,n+1} \end{bmatrix},$$

where

$d_n^T = (d_{n,n-L+1}, d_{n,n-L+2}, ..., d_{n,n-1}) = M\{\xi_{n+1}\Xi_{n|L-1}^T\}$

Since the matrix $D_{n+1|L}$ has a block representation, then

$$D_{n+1|L}^{-1} = \begin{bmatrix} D_{n|L-1}^{-1} + \dfrac{D_{n|L-1}^{-1} d_n d_n^T D_{n|L}^{-1}}{\alpha_{n+1}} & \vdots & -\dfrac{D_{n|L-1}^{-1} d_n}{\alpha_{n+1}} \\ ... & ... & ... \\ -\dfrac{d_n^T D_{n|L-1}^{-1}}{\alpha_{n+1}} & \vdots & \dfrac{1}{\alpha_{n+1}} \end{bmatrix},$$

where $\alpha_{n+1} = d_{n+1,n+1} - d_n^T D_{n|L-1}^{-1} d_n$.

Let's assume that on $n-$ m cycle the following estimate

$$\left(X^T_{n|L} D^{-1}_{n|L} X_{n|L}\right) c_{n|L} = X^T_{n|L} D^{-1}_{n|L} Y_{n|L} \tag{22}$$

is received.

The arrival of new information (adding a new dimension) leads to the calculation of an estimate, which, by analogy with (17), can be written as follows:

$$c_{n+1|L+1} = (X^T_{n+1|L+1} X_{n+1|L+1})^{-1} X^T_{n+1|L+1} Y_{n+1|L+1}, \tag{23}$$

where

$$Y_{n+1|L+1} = \begin{pmatrix} Y_{n-1|L} \\ --- \\ y_{n+1} \end{pmatrix} = \begin{pmatrix} y_{n-L+1} \\ --- \\ Y_{n+1|L} \end{pmatrix} - \text{vector } (L+1)\times 1; \tag{24}$$

$$X_{n+1|L+1} = \begin{pmatrix} X_{n|L} \\ --- \\ x^T_{n+1} \end{pmatrix} = \begin{pmatrix} x^T_{n-L+1} \\ --- \\ X_{n+1|L} \end{pmatrix} - \text{the matrix } (L+1)\times N; \tag{25}$$

Let’s introduce the notation

$$P^{-1}_{n+1|L+1} = \left(X^T_{n+1|L+1} D^{-1}_{n+1|L+1} X_{n+1|L+1}\right);$$

$$P^{-1}_{n|L} = \left(X^T_{n|L} D^{-1}_{n|L} X_{n|L}\right);$$

$$P^{-1}_{n+1|L} = \left(X^T_{n+1|L} D^{-1}_{n+1|L} X_{n+1|L}\right)$$

and calculate $P^{-1}_{n+1|L+1}$

$$P^{-1}_{n+1|L+1} = X^T_{n|L} D^{-1}_{n|L} X_{n|L} + \frac{X^T_{n|L} D^{-1}_{n|L} d_n d^T_n D^{-1}_{n|L} X_{n|L}}{\alpha_{n+1}} - \frac{x_{n+1} d^T_n D^{-1}_{n|L} X_{n+1|L+1}}{\alpha_{n+1}} - \frac{X^T_{n|L} D^{-1}_{n|L} d_n x_{n+1}}{\alpha_{n+1}} +$$

$$+ \frac{x_{n+1} x^T_{n+1}}{\alpha_{n+1}} = P^{-1}_{n|L} + x^*_{n+1} x^{*T}_{n+1},$$

where $x^*_{n+1} = \dfrac{x_{n+1} - X^T_{n|L} D^{-1}_{n|L} d_n}{\sqrt{\alpha_{n+1}}}$

Also similarly calculate

$$X^T_{n+1|L+1} D^{-1}_{n+1|L+1} Y_{n+1|L+1} = X^T_{n|L} D^{-1}_{n|L} Y_{n|L} + x^*_{n+1} y^*_{n+1},$$

where $y^*_{n+1} = \dfrac{y_{n+1} - Y_{n|L} D^{-1}_{n|L} d_n}{\sqrt{\alpha_{n+1}}}.$

Adding to both parts of (22) $x^*_{n+1} x^{*T}_{n+1} c_{n|L}$

$$P^{-1}_{n|L} c_{n|L} + x^*_{n+1} x^{*T}_{n+1} c_{n|L} = X^T_{n|L} D^{-1}_{n|L} Y_{n|L} + x^*_{n+1} x^{*T}_{n+1} c_{n|L}$$

and subtracting (22) from (23) (taking into account the properties $P^{-1}_{n|L}$ and $X^T_{n+1|L+1} D^{-1}_{n+1|L+1} Y_{n+1|L+1}$) we receive

$$P^{-1}_{n+1|L+1}\left(c_{n+1|L+1} - c_{n|L}\right) = x^*_{n+1}\left(y^*_{n+1} - c^T_{n|L} x^*_{n+1}\right)$$

or

$$c_{n+1|L+1} = c_{n|L} + P_{n+1|L+1} x^*_{n+1}\left(y^*_{n+1} - c^T_{n|L} x^*_{n+1}\right),$$

where

$$P_{n+1|L+1} = P_{n|L} - \frac{P_{n|L} x^*_{n+1} x^{*T}_{n+1} P_{n|L}}{1 + x^{*T}_{n+1} P_{n|L} x^*_{n+1}}.$$

When discarding outdated information received at n – L + 1 step, we come from evaluation $c_{n+1|L+1}$ to the assessment $c_{n+1|L}$ ... To obtain the corresponding rules for correcting the estimate, we will proceed as follows.

We use the block representation of the covariance matrix $D_{n+1|L+1}$

$$D_{n+1|L+1} = \begin{bmatrix} d_{n-L+1,n-L+1} & d_{n-L+1,n-L+21}\cdots & d_{n-L+1,n} & d_{n-L+1,n+1} \\ d_{n-L+2,n-L+1} & d_{n-L+2,n-L+2} & d_{n-L+2,n} & d_{n-L+1,n+1} \\ \cdots & \cdots & \cdots & \cdots \\ d_{n+1,n-L+1} & d_{n+1,n-L+1} & d_{n+1,n} & d_{n+1,n+1} \end{bmatrix} = \begin{bmatrix} d_{n-L+1,n-L+1} & \vdots & d_{n-L+1}^T \\ - & - & - \\ d_{n-L+1} & \vdots & D_{n+1|L} \end{bmatrix},$$

where

$$d_{n-L+1}^T = \left(d_{n-L+1,n-L+2}, d_{n-L,n-L+3},\ldots, d_{n-L+1,n+1}\right) = M\left\{\xi_{n-L+1}\Xi_{n+1|L}^T\right\},$$

and the inverse matrix representation $D_{n+1|L+1}^{-1}$ as

$$D_{n+1|L+1}^{-1} = \begin{bmatrix} \frac{1}{\alpha_{n-L+1}} & \vdots & -\frac{d_{n-L+1}^T D_{n+1|L}^{-1}}{\alpha_{n-L+1}} \\ \cdots & \cdots & \cdots \\ -\frac{D_{n+1|L}^{-1} d_{n-L+1}}{\alpha_{n-L+1}} & \vdots & D_{n+1|L}^{-1} + \frac{D_{n+1|L}^{-1} d_{n-L+1} d_{n-L+1}^T D_{n+1|L}^{-1}}{\alpha_{n-L+1}} \end{bmatrix},$$

where $\alpha_{n-L+1} = d_{n-L+1,n-L+1} - d_{n-L+1}^T D_{n+1|L}^{-1} d_{n-L+1}$.

In this case

$$P_{n+1|L+1}^{-1} = \left(X_{n+1|L+1}^T D_{n+1|L+1}^{-1} X_{n+1|L+1}\right) = \frac{x_{n-L+1} x_{n-L+1}^T}{\alpha_{n-L+1}} - \frac{X_{n+1|L}^T D_{n+1|L}^{-1} d_{n-L+1} d_{n-L+1}^T D_{n+1|L}^{-1} X_{n+1|L}}{\alpha_{n-L+1}} -$$

$$- \frac{x_{n-L+1} d_{n-L+1}^T D_{n+1|L}^{-1} X_{n+1|L}}{\alpha_{n-L+1}} - \frac{X_{n+1|L}^T D_{n+1|L}^{-1} d_{n-L+1} x_{n-L+1}^T}{\alpha_{n-L+1}} + X_{n+1|L}^T D_{n+1|L}^{-1} X_{n+1|L} =$$

$$= P_{n+1|L}^{-1} + x_{n-L+1}^* x_{n-L+1}^{*T},$$

where

$$x_{n-L+1}^* = \frac{x_{n-L+1} - X_{n+1|L}^T D_{n+1|L}^{-1} d_{n-L+1}}{\sqrt{\alpha_{n-L+1}}}.$$

Similarly

$$X_{n+1|L+1}^T D_{n+1|L+1}^{-1} Y_{n+1|L+1} = X_{n+1|L}^T D_{n+1|L}^{-1} Y_{n+1|L} + x_{n-L+1}^* y_{n-L+1}^*,$$

where $y_{n-L+1}^* = \frac{y_{n-L+1} - Y_{n+1|L} D_{n+1|L}^{-1} d_{n-L+1}}{\sqrt{\alpha_{n-L+1}}}$.

Subtraction from both parts of (23) $x_{n-L+1}^* x_{n-L+1}^{*T} c_{n+1|L+1}$... gives

$$\left(\left(X_{n+1|L+1}^T D_{n+1|L+1}^{-1} Y_{n+1|L+1}\right)^{-1} - x_{n-L+1}^* x_{n-L+1}^{*T}\right) c_{n+1|L+1} = X_{n+1|L+1}^T D_{n+1|L+1}^{-1} Y_{n+1|L+1} - x_{n-L+1}^* x_{n-L+1}^{*T} c_{n+1|L+1}.$$

Considering that

$$\left(X_{n+1|L}^T D_{n+1|L}^{-1} X_{n+1|L}\right) c_{n+1|L} = X_{n+1|L}^T D_{n+1|L}^{-1} Y_{n+1|L}, \qquad (26)$$

subtraction from (26) of relation (23) (taking into account the expressions for $P_{n+1|L}^{-1}$ and $X_{n+1|L}^T D_{n+1|L}^{-1} Y_{n+1|L}$)

$$P_{n+1|L}^{-1}\left(c_{n+1|L} - c_{n+1L+1}\right) = x_{n-L+1}^* x_{n-L+1}^{*T} c_{n+1|L+1} - x_{n-L+1}^* y_{n-L+1}^*,$$

from where

$$c_{n+1|L} = c_{n+1|L+1} - P_{n+1|L}x^*_{n-L+1}\left(y^*_{n-L+1} - c^T_{n+1|L+1}x^*_{n-L+1}\right),$$

but

$$P^{-1}_{n+1|L} = P^{-1}_{n+1|L+1} - x^*_{n-L+1}x^{*T}_{n-L+1},$$

therefore

$$P_{n+1|L} = P_{n+1|L+1} + \frac{P_{n+1|L+1}x^*_{n-L+1}x^{*T}_{n-L+1}P_{n+1|L+1}}{1 - x^{*T}_{n-L+1}P_{n+1|L+1}x^*_{n-L+1}}.$$

Thus, the algorithm will have the form (the first two relations describe the inclusion of newly arrived information, and the next ones describe the discarding of outdated information)

$$c_{n+1|L+1} = c_{n|L} + P_{n+1|L+1}x^*_{n+1}\left(y^*_{n+1} - c^T_{n|L}x^*_{n+1}\right), \tag{27}$$

$$P_{n+1|L+1} = P_{n|L} - \frac{P_{n|L}x^*_{n+1}x^{*T}_{n+1}P_{n|L}}{1 + x^{*T}_{n+1}P_{n|L}x^*_{n+1}}. \tag{28}$$

$$c_{n+1L} = c_{n+1|L+1} - P_{n+1|L}x^*_{n-L+1}\left(y^*_{n-L+1} - c^T_{n+1|L+1}x^*_{n-L+1}\right), \tag{29}$$

$$P_{n+1|L} = P_{n+1|L+1} + \frac{P_{n+1|L+1}x^*_{n-L+1}x^{*T}_{n-L+1}P_{n+1|L+1}}{1 - x^{*T}_{n-L+1}P_{n+1|L+1}x^*_{n-L+1}}. \tag{30}$$

If at first outdated information is discarded, and then the newly received information is included, then the algorithm takes the form

$$c_{n+1|L-1} = c_{n+1|L} - P_{n|L-1}x^*_{n-L+1}\left(y^*_{n-L+1} - c^T_{n+1|L}x^*_{n-L+1}\right), \tag{31}$$

$$P_{n|L-1} = P_{n|L} + \frac{P_{n|L}x^*_{n-L+1}x^{*T}_{n-L+1}P_{n|L}}{1 - x^{*T}_{n-L+1}P_{n|L}x^*_{n-L+1}}, \tag{32}$$

$$c_{n+1|L} = c_{n+1|L-1} + P_{n+1|L}x^*_{n+1}\left(y^*_{n+1} - c^T_{n-1|L-1}x^*_{n+1}\right), \tag{33}$$

$$P_{n+1|L} = P_{n|L-1} - \frac{P_{n|L-1}x^*_{n+1}x^{*T}_{n+1}P_{n|L-1}}{1 + x^{*T}_{n+1}P_{n|L-1}x^*_{n+1}}; \tag{34}$$

where

$$x^*_{n-L+1} = \frac{x_{n-L+1} - X^T_{n+1|L}D^{-1}_{n+1|L}d_{n-L+1}}{\sqrt{\alpha_{n-L+1}}};$$

$$x^*_{n+1} = \frac{x_{n+1} - X^T_{n|L}D^{-1}_{n|L}d_n}{\sqrt{\alpha_{n+1}}}; \tag{35}$$

$$y^*_{n-L+1} = \frac{y_{n-L+1} - Y_{n+1|L}D^{-1}_{n+1|L}d_{n-L+1}}{\sqrt{\alpha_{n-L+1}}};$$

$$y^*_{n+1} = \frac{y_{n+1} - Y_{n|L}D^{-1}_{n|L}d_n}{\sqrt{\alpha_{n+1}}}.$$

5. Recurrent TPA algorithm in the presence of outliers and correlated noise

As noted above, the current regression analysis algorithm, which has the form (5), allows two forms of presenting estimates, due to the order of using information about newly received measurements and the oldest ones.

Let's dwell on this in more detail.

Obtaining new information (adding a new dimension) leads to the calculation of an estimate, which can be written in the form (23)

Since at each cycle, when constructing an estimate, $L = const$, then consider the case when new dimensions are added first, and then obsolete ones are excluded.

The recurrent form of estimate (23) can be obtained by standard methods using the block representation of vectors and matrices (24), (25), which allows rewriting (23) as follows:

$$c_{n+1|L} = (X_{n|L}^T X_{n|L} + x_{n+1}x_{n+1}^T - x_{n-L+1}x_{n-L+1}^T)^{-1}(x_{n-L+1}X_{n|L}^T \vdots x_{n+1})\begin{bmatrix} y_{n-L+1} \\ Y_{n|L} \\ --- \\ y_{n+1} \\ \end{bmatrix}. \qquad (36)$$

Let us consider a modification of the current regression analysis algorithm used to maximize the correlation (12) and which, unlike (36), will have the form

$$c_{n+1|L} = (X_{n|L}^T X_{n|L} + \psi_{n+1}x_{n+1}^* x_{n+1}^{*T} - \psi_{n-L+1}x_{n-L+1}^* x_{n-L+1}^{*T})^{-1}(x_{n-L+1}^* X_{n|L}^T \vdots x_{n+1}^*)\begin{bmatrix} y_{n-L+1}^* \\ Y_{n|L} \\ --- \\ y_{n+1}^* \\ \end{bmatrix}.$$

By designating

$$P_{n+1|L+1}^{-1} = X_{n+1|L+1}^T X_{n+1|L+1};$$

$$P_{n|L}^{-1} = X_{n|L}^T X_{n|L}$$

and taking into account (24), (25), we have

$$P_{n+1|L+1}^{-1} = P_{n|L}^{-1} + \psi_{n+1}x_{n+1}^* x_{n+1}^{*T} - \psi_{n-L+1}x_{n-L+1}^* x_{n-L+1}^{*T}. \qquad (37)$$

Applying the matrix inversion lemma to (37), we can obtain, as already noted, two forms of computations: in one, the accumulation of information is used first (the newly arrived signal x_{n+1}), and then outdated information is discarded (signal x_{n-L+1}) and vice versa. So the calculation of the matrix and the refinement of estimates when accumulating information occurs, respectively, according to the formulas

$$P_{n+1|L+1} = P_{n|L} - \frac{\psi_{n+1}P_{n|L}x_{n+1}^* x_{n+1}^{*T}P_{n|L}}{1+\psi_{n+1}x_{n+1}^{*T}P_{n|L}x_{n+1}^*}. \qquad (38)$$

$$c_{n+1|L+1} = c_{n|L} + \frac{\psi_{n+1}P_{n|L}x_{n+1}^*}{1+\psi_{n+1}x_{n+1}^{*T}P_{n|L}x_{n+1}^*}(y_{n+1} - c_{n|L}^T x_{n+1}^*). \qquad (39)$$

Ratios corresponding to the discarding of obsolete information due to the fact that

$$P_{n+1|L}^{-1} = X_{n+1|L}^T X_{n+1|L} = P_{n+1|L+1}^{-1} - \psi_{n-L+1}x_{n-L+1}^* x_{n-L+1}^{*T}$$

looks like

$$P_{n+1|L} = P_{n+1|L+1} + \frac{\psi_{n-L+1}P_{n+1|L+1}x_{n-L+1}^* x_{n-L+1}^{*T}P_{n+1|L+1}}{1-\psi_{n-L+1}x_{n-L+1}^{*T}P_{n+1|L+1}x_{n-L+1}^*}; \qquad (40)$$

$$c_{n+1|L} = c_{n+1|L+1} - -\frac{\psi_{n-L+1}P_{n+1|L+1}x_{n-L+1}^*}{1-\psi_{n-L+1}x_{n-L+1}^{*T}P_{n+1|L+1}x_{n-L+1}^*}(y_{n-L+1}^* - c_{n+1|L+1}^T x_{n-L+1}^*). \qquad (41)$$

Thus, the recurrent estimation algorithm obtained by adding new information and then excluding obsolete information is described by relations (38) – (41).

6. Parameter σ selection

There are many ways to choose the optimal kernel size. One of the most commonly used methods of choosing an appropriate kernel width in machine learning is cross validation. Another fairly simple approach is the Silverman's rule of thumb [16]

$$\sigma = 0{,}9AN^{-1/5}, \tag{42}$$

where A is the smallest value between the standard deviation of the data sample and the interquartile range of the data, scaled by 1.34, and N is the number of data samples.

As can be seen from (10), the cost function (criterion) of algorithms based on correntropy changes depending on the width σ, the size of which affects the accuracy of the estimate. Since the reference signals change at random, this leads to the need to apply a time-varying kernel size.

The rule of thumb proposed by Silverman was applied in [17] as follows:

$$\sigma = \left(\frac{4}{3n}\right)^{1/5}\hat{\sigma}, \tag{43}$$

$$\hat{\sigma} = \frac{1}{L-1}\left(\sum_{i=1}^{L} x_i^2 - L\bar{x}^2\right), \tag{44}$$

where $\hat{\sigma}$ denotes the variance of the signal sample.

These relations were used in [18] to recursively update the kernel size based on the sample variance using the formula

$$\sigma_{n+1}^2 = \gamma\sigma_n^2 + (1-\gamma)\hat{\sigma}, \tag{45}$$

where $\gamma(0 << \gamma < 1)$ is close to 1, and $\hat{\sigma}$ is a sample of the variance of the reference signal x_n... Since the value σ_n^2 proportional to the variance of the control sample, a noisy pulse standard can cause a large σ_n^2, which weakens the stability of the algorithm. Therefore, in this work, the threshold ψ is set for σ_n^2

$$\sigma_n^2 \le \psi, \tag{46}$$

where ψ is determined based on the real situation.

In [19], an algorithm for adaptive changes in the kernel width is proposed, which is based on the analysis of the following rule:

$$\sigma = \frac{\max|e_i|}{2\sqrt{2}}, \qquad i = 1,2,...N. \tag{47}$$

When choosing a variable σ, the function $f(\sigma^2)$ will also be variable and therefore the rule for updating the weights can be controlled.

In [20], the case of correction σ under the assumption that the kernel width linearly depends on the instantaneous error, i.e.

$$\sigma_{n+1} = k_\sigma|e_{n+1}|, \tag{48}$$

where k_σ is a positive constant.

In [21], it is proposed to use the following function in the estimation algorithm $f(\sigma^2)$:

$$f(\sigma^2) = \frac{\alpha\left(1-\exp\left(-\sigma^2\right)\right)}{\sqrt{1+\left(\frac{\sigma^2}{B}\right)^{2m}}}. \tag{49}$$

This function has all the necessary properties and is based on the Butterworth filter. In (11) α – gain; m and B – filter order and throughput, respectively. Options α and m can be either fixed or adaptively changeable. Since the bandwidth B is another parameter that significantly affects $f(\bullet)$, an

attempt is made to adapt it at each iteration based on the analysis of the error $|e_{n+1}|$ The quantity B determines whether σ_{n+1} outlier or not. Therefore, in this work as B at time n the average of all past error patterns is selected. Such choice of B allows to reduce the influence of outliers of sampling errors and leads to a slowdown in the rate of convergence of the estimation algorithm.

In [13], to determine the optimal value of the variable σ_n the optimization problem is solved. To maximize $f(\sigma_n)$ the derivative of (11) with respect to σ_n equals to zero

$$\frac{e_{n+1}^2 - \xi_{n+1}^2 - e_{n+1}^a \xi_{n+1}}{\gamma \|x_{n+1}\|^2 e_{n+1}^2} = \exp\left(-\frac{e_{n+1}^2}{2\sigma_{n+1}^2}\right), \tag{50}$$

which produces the following expression:

$$\sigma_{n+1}^2 = \frac{-e_{n+1}^2}{2\ln\left[\frac{e_{n+1}^2 - \xi_{n+1}^2 - e_{n+1}^a \xi_{n+1}}{\gamma \|x_{n+1}\|^2 e_{n+1}^2}\right]}. \tag{51}$$

Here $e_{n+1}^a = \theta_n^T x_{n+1}$ – a priori error; $e_{n+1} = e_{n+1}^a + \xi_{n+1}, ...$

Since the information about the implementation of the noise ξ_{n+1} usually absent, it is not possible to use this formula. Therefore, for the practical application of the correction rule σ_{n+1}^2 in this paper it is proposed to replace ξ_{n+1}^2 with noise variance σ_ξ^2 and furthermore, it is assumed that the prior error e_{n+1}^a, does not depend on noise ξ_{n+1}, i.e. it is assumed that $M\{e_{n+1}^a \xi_{n+1}\} = 0$. As noted in this paper, the introduction of the approximation $e_{n+1}^a \xi_{n+1} \approx 0$ is quite reasonable, since on average this product is zero. Thus, in the final form, the correction rule σ_{n+1}^2 has following form:

$$\sigma_{n+1}^2 = \frac{-e_{n+1}^2}{2\ln\left[\frac{e_{n+1}^2 - \sigma_\xi^2}{\gamma \|x_{n+1}\|^2 e_{n+1}^2}\right]}. \tag{52}$$

For a smooth update σ_{n+1}^2 using the moving average method [22], the following rule is proposed in the work:

$$\sigma_{n+1}^2 = \begin{cases} \alpha\sigma_n^2 + (1-\alpha)\min\left(\frac{-e_{n+1}^2}{2\ln \chi_{n+1}}, \sigma_n^2\right), & if \quad 0 < \chi_{n+1} < 1, \\ \sigma_n^2 \quad otherwise, \end{cases} \tag{53}$$

where α – smoothing coefficient close to one, and

$$\chi_{n+1} = \frac{e_{n+1}^2 - \sigma_\xi^2}{\gamma \|x_{n+1}\|^2 e_{n+1}^2}. \tag{54}$$

As seen in (53), to provide a positive square kernel width σ_{n+1}^2 the suggested kernel value is updated when $0 < \chi_{n+1} < 1$. In addition, it can be seen from (53) that in the update σ_{n+1}^2 plays a major role χ_{n+1}, which, as follows from (54), depends on the values e_{n+1}^2, $\|x_{n+1}\|^2$ and σ_ξ^2 ... In the case of noise with time-varying characteristics, the learning strategy described in [23] can be used to estimate the time-varying noise variance. Thus, the approach proposed in this paper is applicable to non-stationary noise as well.

In [24], a modification of the RLMS is proposed, supplemented by an online recursive scheme for adapting the kernel size, using the analysis of error values on a number of observations

$$m_{\sigma,n+1} = m_{\sigma,n} + \Delta m_{\sigma,n+1} \tag{55}$$

where

$$\Delta m_{\sigma,n+1} = \frac{1}{N_w}\left[e_n - e_{n-N_w+1}\right] \tag{56}$$

Here N_w is the size of the observation window. In the paper, e_n is estimated rather roughly using only the manifold of the window's edge.

In [25], the following correction scheme is proposed for σ^2_{n+1}:

$$\sigma^2_{n+1} = \sigma^2_n + \Delta m^2_{\sigma,n+1} + \underbrace{\frac{1}{N_w}\left[e_n - m_{\sigma,n+1}\right]^2}_{I} - \underbrace{\frac{1}{N_w}\left[e_{n-N_w+1} - m_{\sigma,n+1}\right]^2}_{II}. \tag{57}$$

It should be noted that terms I and II can be considered as compensation for estimating e_n... To reduce the computational load, this expression can be simplified as follows:

$$\sigma^2_{n+1} = \sigma^2_n + \Delta m^2_{\sigma,n+1}. \tag{58}$$

Analysis of the above approaches to parameter selection σ shows that there is no single rule for choosing this parameter; therefore, in the practical implementation of algorithms based on maximizing the correlation, one should be guided by the recommendations discussed above.

7. Conclusion

In this work, the main relations that describe an adaptive multi-step algorithm for training ADALINA are obtained, which allows to adjust its parameters in real time in the presence of outliers and correlated noise. The use of such an algorithm accelerates the learning process by using information not only about one last cycle (as in the traditional Widrow-Hoff learning algorithm), but also about a number of previous cycles. The robustness of the estimates is ensured by the application of the maximum correlation criterion.

8. Acknowledgements

The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

9. References

[1] B. Widrow, M. Hoff, Adaptive switching circuits, IRE WESCON Convention Record. Part 4. New York: Institute of Radio Engineers, 1960, p. 96–104.

[2] B.D. Liberol, O.G. Rudenko, A.A. Bessonov, Investigation of the convergence of one-step adaptive identification algorithms, Problems of Control and Informatics, 2018.5, pp. 19–32.

[3] O.G. Rudenko, A.A. Bessonov, Regularized algorithm for learning adalina in the problem of estimating non-stationary parameters,Control systems and machines, 2019.1, pp.22–30.

[4] P. Huber, Robustness in statistics. – M.: Mir, 1984, 304 p.

[5] F.R. Hampel, E.M. Ronchetti, P.J. Rousseeuw, W.A. Stahel, Robust Statistics. The Approach Based on Influence Functions. – NY: John Wiley and Sons, 1986, 526 p.

[6] I. Santamaría, P.P. Pokharel, C. Jose, J.C. Principe, Generalized Correlation Function: Definition, Properties, and Application to Blind Equalization, IEEE Trans. on Signal Processing, Vol. 54, no. 6, 2006, pp. 2187–2197. DOI:10.1109 / TSP.2006.872524

[7] W. Liu, P.P. Pokharel, J.C. Principe, Correntropy: Properties and Applications in Non-Gaussian Signal Processing, IEEE Trans. on Signal Processing, 1, 2007, pp. 5286–5298DOI: 10.1109 / TSP.2007.896065

[8] W. Wang, J. Zhao, H. Qu, B. Chen, J.C. Principe, An adaptive kernel width update method of correntropy for channel estimation, IEEE International Conference on Digital Signal Processing (DSP), 2015, pp. 916–920. DOI:10.1109 / ICDSP.2015.7252010
[9] A. Gunduz, J.C. Principe, Correntropy as a novel measure for nonlinearity tests / Signal Processing, 2009, v. 89, pp. 14–23. URL: https://doi.org/10.1016/j.sigpro.2008.07.005
[10] Y. Guo, B. Ma, Y. Li, Kernel-Width Adaption Diffusion Maximum Correntropy Algorithm / IEEE Acces, 2016, v. 4, pp.1–14. DOI: 10.1109 / ACCESS.2020.2972905. URL: https://doi.org/10.36227/techrxiv.11842281.v1
[11] L. Lu, H. Zhao, Active impulsive noise control using maximum correntropy withadaptive kernel size, Mechanical Systems and Signal Processing, 2017, v. 87, Part A., pp. 180–191. URL: https://doi.org/10.1016/j.ymssp.2016.10.020
[12] Y. Qi, Y. Wang, J. Zhang, J. Zhu, X. Zheng, Robust Deep Network with Maximum Correntropy Criterion for Seizure Detection, BioMed Research International. Volume 2014, Article ID 703816, 10 p. URL:http://dx.doi.org/10.1155/2014/703816
[13] L. Shi, H. Zhao, Y. Zakharov, An Improved Variable Kernel Width for Maximum Correntropy Criterion Algorithm, IEEE Trans. on Circuits and Systems II: Express Briefs, 2018, 5p. DOI: 10.1109 / TCSII.2018.2880564
[14] I.I. Perelman, Operational identification of control objects, M .: Energoizdat, 1982, 272 p.
[15] O.G. Rudenko, I.D. Terenkovsky, A. Shtefan, G.A. Oda, Modified algorithm of the current regression analysis in identification and forecasting problems, Radioelectronics and Informatics, 1998, No. 4 (05), pp. 58–61.
[16] B.W. Silverman, Density Estimation for Statistics and Data Analysis, vol. 3: CRC Press: New York, NY, USA, 1986, 176 p.
[17] W. Wertz, Statistical Density estimation: A survey, Goettingen: Vandenhoeck and Ruprecht, 1978, 108 p.
[18] Z.C. Hea, H.H. Yea, E. Lib, An efficient algorithm for Nonlinear Active Noise Control of Impulsive Noise, Applied Acoustics, 2019, Vol. 148, pp. 366–374.
[19] Y. Liu, J. Chen Correntropy-based kernel learning for nonlinear system identification with unknown noise: an industrial case study, Proc. of the 10th IFAC Symposium on Dynamics and Control of Process Systems, 2013, pp. 361–366.
[20] J.C. Munoz, J.H. Chen, Removal of the effects of outliers in batch process data through maximum correntropy estimator, Chemom. Intell. Lab. Syst., 2012, pp. 53–58.
[21] F. Huang, J. Zhang, S. Zhang Adaptive filtering under a variable kernel width maximum correntropy criterion, IEEE Transactions on Circuits and Systems II: Express Briefs, 2017, Vol. 64, no.10, pp. 1247–1251.
[22] L. Lu, H. Zhao, Active impulsive noise control using maximum correntropy with adaptive kernel size, Mechanical Systems and Signal Processing, 2017, vol. 87, pp. 180–191.
[23] M. Bergamasco, F.D. Rossa, L. Piroddi, Active noise control with on-line estimation of non-Gaussian noise characteristics, J. Sound Vib., 2012, 331 (1), pp. 27–40.
[24] M. Belge, E.L. Miller, A sliding window RLS-like adaptive algorithm for filtering alpha-stable noise, IEEE Signal Process. Lett., 2000, vol. 7., pp. 86–89.
[25] A.N. Vazquez, J.A. Garcia, Combination of recursive least-norm algorithms for robust adaptive filtering in alpha-stable noise, IEEE Trans. Signal Process, 2012, vol. 60 (3), pp. 1478–1482.