THE COMPARISON OF THE MONTE-CARLO METHOD AND NEURAL NETWORKS ALGORITHMS IN NONLINEAR ESTIMATION PROBLEMS

O.A. Stepanov*, O.S. Amosov**

**State Research Center of Russia - Central Scientific & Research Institute Elektropribor*
30, Malaya Posadskaya Str., St. Petersburg, 197046, Russia
Tel. (812) 232-82-53, fax (812) 232-33-76, e-mail: elprib@online.ru
***Komsomolsk-on-Amur State Technical University*
27, Lenina Str., Komsomolsk-on-Amur, 681013, Russia
Tel. (4217) 55-41-51, fax (4217) 53-61-50, e-mail: aos@kmscom.ru

Abstract: The paper compares the algorithms based on neural networks and the Monte-Carlo method as applied to nonlinear estimation problems solved in the framework of the Bayesian approach. Two variants are considered. The first variant is a search of optimal estimates that are conditional mathematical expectations and, in a general case, depend on measurements in a nonlinear way. The second variant involves linear optimal estimates. In designing them, the root-mean-square criterion is minimized in the class of estimates that are linearly dependent on measurements. The comparison results are discussed.

Keywords: algorithm, nonlinear estimation, minimum variance estimate, neural networks, Monte-Carlo method, comparison, navigation problems.

1. INTRODUCTION

In practice it is rather common that the time-invariant vector needs to be estimated by the measurements whose dependence on this vector is nonlinear. The problems like this are, for instance, to be solved in navigational data processing (Stepanov, 1998; Bergman, 1999). In solving this problem the theory of optimal estimation based on the Bayesian approach is used. The latter allows finding optimal (minimum variance) estimates as an a posteriori mathematical expectation (corresponding to the a posteriori probability density function – p.d.f.) of the vector under estimation. If the problem is nonlinear, the a posteriori p.d.f. is not Gaussian. This causes the need for designing efficient suboptimal algorithms that do not involve a large size (quantity) of calculations. There are various suboptimal algorithms: the point mass method; the Monte-Carlo method and its modifications (particle filters, for example), the method based on poly-Gaussian, and various Gaussian approximations of a posteriori p.d.f., etc. (Jazvinski, 1970; Stepanov, 1998; Bergman, 1999; Daum, 2005). Besides, considerable recent attention has been focused on the possibilities of using artificial neural networks (NN), in particular, the so-called supervised learning NN (Haykin, 1994, 2001, Stepanov and Amosov 2006). A distinguishing feature of the supervised NN is the fact that it needs a training set for learning NN. However, for many applied estimation problems (for example, in solving navigation problems) it is impossible to obtain a training set from experimental data. Nevertheless, in these conditions the training set can be formed by simulation, using prescribed models. Thus the NN can be considered as a numerical procedure for calculation of Bayesian estimates. The purpose of using NN in this case is to develop economical, from the computational standpoint, procedures for calculation of optimal estimates. In this connection it is useful to compare algorithms based on NN with those that are traditionally used for the solution of nonlinear estimation problems. In so doing, the Monte-Carlo

method, widely used recently, was taken as a traditional algorithm (Zaritsky, *et al.,* 1975, Haykin, 2001). The choice of this method is due to the fact that it, just as NN, uses realizations of random vectors (samples) derived by simulation in accordance with the known models.

The paper analyzes the features of NN algorithms in comparison with those based on the Monte-Carlo method for two variants of the solution. The first variant is aimed at searching optimal estimates, representing the a posteriori mathematical expectations that are, in a general case, dependent on measurements in a nonlinear way. The second case involves linear optimal estimates. The results obtained are illustrated by the example of the solution of a nonlinear problem of determining a vehicle's position by reference beacons.

2. NONLINEAR ESTIMATION PROBLEM

Let us consider the following problem: to estimate an n-dimensional random vector $\mathbf{x} = [x_1 \quad ... \quad x_n]^T$ by m- dimensional measurements $\mathbf{y} = [y_1 \quad ... \quad y_m]^T$

$$\mathbf{y} = \mathbf{s}(\mathbf{x}) + \mathbf{v}, \qquad (1)$$

where $\mathbf{s}(\mathbf{x}) = [s_1(\mathbf{x}) \quad ... \quad s_m(\mathbf{x})]^T$ is the known m-dimensional nonlinear vector-function; $\mathbf{v} = [v_1 \quad ... \quad v_m]^T$ is a random vector of measurement errors. Suppose that the joint p.d.f. $f(\mathbf{x}, \mathbf{v})$ for the vectors $\mathbf{x}$ and $\mathbf{v}$ is known. For simplicity $\mathbf{v}$ and $\mathbf{x}$ are assumed to be zero mean random vectors independent of each other, i.e. $f(\mathbf{x}, \mathbf{v}) = f(\mathbf{x}) f(\mathbf{v})$. Thus, taking into account (1), it is possible to get the p.d.f. $f(\mathbf{x}, \mathbf{y})$ for $\mathbf{x}$ and $\mathbf{y}$ $f(\mathbf{x}, \mathbf{y}) = f(\mathbf{x}) f_{\mathbf{v}}(\mathbf{y} - \mathbf{s}(\mathbf{x}))$.

The assumptions made allow stating the problem of finding the optimal (minimum variance) estimate $\hat{\mathbf{x}}(\mathbf{y})$ that minimizes the criterion

$$J = E[\|(\mathbf{x} - \hat{\mathbf{x}}(\mathbf{y})\|^2], \qquad (2)$$

where $\|a\|^2 = a^T a$; E is the mathematical expectation corresponding to $f(\mathbf{x}, \mathbf{y})$.

Two variants of the solution to this problem are discussed below.

2.1 Nonlinear optimal estimate

It is known that the optimal estimate and the covariance matrix of estimation errors are determined as (Jazwinski, 1970)

$$\hat{\mathbf{x}}^{opt}(\mathbf{y}) = \int \mathbf{x} f(\mathbf{x}/\mathbf{y}) d\mathbf{x}, \qquad (3)$$

$$\mathbf{P}^{opt}(\mathbf{y}) = \int (\mathbf{x} - \hat{\mathbf{x}}^{opt}(\mathbf{y}))(\mathbf{x} - \hat{\mathbf{x}}^{opt}(\mathbf{y}))^T f(\mathbf{x}/\mathbf{y}) d\mathbf{x}, \qquad (4)$$

where $f(\mathbf{x}/\mathbf{y})$ is the a posteriori (conditional) p.d.f. for the vector $\mathbf{x}$. It should be noted that the symbol of the integral in these expressions and below corresponds to the multiple integrals with infinite limits. The matrix $\mathbf{P}^{opt}(\mathbf{y})$ (conditional error covariance matrix) characterizes the accuracy of the state-vector estimate for the given set of the measurements $\mathbf{y}$.

It is well known that the problem of designing an algorithm for the calculation of $\hat{\mathbf{x}}^{opt}(\mathbf{y})$ and $\mathbf{P}^{opt}(\mathbf{y})$ is easily solved only for the Gaussian $f(\mathbf{x}, \mathbf{v})$ and the linear character of the function $\mathbf{s}(\mathbf{x})$, i.e. when $\mathbf{s}(\mathbf{x}) = \mathbf{H}\mathbf{x}$. For this case the dependence of the optimal estimate on measurements has a linear character. In all other cases there arises the problem of designing suboptimal algorithms that do not involve a large size of calculations. As, in general, the dependence of the minimum variance estimate on measurements is nonlinear, for convenience, below the estimate (3) will be called a nonlinear optimal estimate.

2.2 Linear optimal estimate

One of the variants of designing suboptimal algorithms is reduced to finding a linear optimal estimate instead of (3). Then the estimate is calculated as $\hat{\mathbf{x}}^{lin}(\mathbf{y}) = \bar{\mathbf{x}} + \mathbf{K}^{lin}[\mathbf{y} - \mathbf{y}^{lin}]$. The idea of designing a linear optimal algorithm consists in choosing the gain factor matrix $\mathbf{K}^{lin}$ and the vector $\mathbf{y}^{lin}$ in such a way as to minimize the criterion (2) in the class of linear estimates. It can be shown that the linear optimal estimate is determined as (Medich, 1969):

$$\hat{\mathbf{x}}^{lin}(\mathbf{y}) = \bar{\mathbf{x}} + \mathbf{P}_{\mathbf{xy}}\mathbf{P}_{\mathbf{yy}}^{-1}[\mathbf{y} - \mathbf{y}^{lin}], \qquad (5)$$

$$\mathbf{K}^{lin} = \mathbf{P}_{\mathbf{xy}}\mathbf{P}_{\mathbf{yy}}^{-1}, \qquad (6)$$

$$\mathbf{P}^{lin} = E[(\mathbf{x} - \hat{\mathbf{x}}(\mathbf{y}))(\mathbf{x} - \hat{\mathbf{x}}(\mathbf{y}))^T] = \mathbf{P}_0 - \mathbf{P}_{\mathbf{xy}}\mathbf{P}_{\mathbf{yy}}^{-1}\mathbf{P}_{\mathbf{yx}}, \qquad (7)$$

where $\bar{\mathbf{x}}$, $\mathbf{y}^{lin} = \bar{\mathbf{y}}$, $\mathbf{P}_0$, $\mathbf{P}_{\mathbf{yy}}$ are the mathematical expectations and the covariance matrices of the vectors $\mathbf{x}$ and $\mathbf{y}$, $\mathbf{P}_{\mathbf{xy}}$ is the cross covariance matrix for $\mathbf{x}$ and $\mathbf{y}$. Thus the problem of finding the optimal linear estimate is reduced to calculation of the first two moments of the joint vector that includes the vector of the parameters being estimated $\mathbf{x}$ and the measurement vector $\mathbf{y}$. After these moments have been derived, the relations (5)-(7) are used to calculate the estimates and the corresponding covariance matrix.

2.3 The aim of the study

The aim of the study is to analyze the possibilities of

using algorithms based on supervised NN for deriving optimal nonlinear and linear estimates and to discuss their peculiarities in comparison with algorithms based on the Monte-Carlo method.

Next let us assume that $\mathbf{x}$ and $\mathbf{v}$ are the Gaussian vectors with the mathematical expectations $\overline{\mathbf{x}} = 0$ and $\overline{\mathbf{v}} = 0$ and the covariance matrices $\mathbf{P}_0$ and $\mathbf{P}_\mathbf{v}$, i.e. $f(\mathbf{x}) = N(\mathbf{x};0,\mathbf{P}_0)$, $f(\mathbf{v}) = N(\mathbf{v};0,\mathbf{P}_\mathbf{v})$.

3. MONTE-CARLO METHOD

As is known, the Monte-Carlo method is convenient to use when it is required to calculate integrals that represent mathematical expectations corresponding to certain known p.d.f. It is precisely this situation that takes place in calculating optimal linear and nonlinear estimates. The advantage of this algorithm is the fact that the accuracy of calculations increases as the number of samples grows. Let us specify the corresponding algorithms and note their peculiarities.

3.1 Nonlinear optimal estimate

For the optimal estimate (3) and covariance (4) the following equations hold true (Zaritsky, *et al.,* 1975)

$$\hat{\mathbf{x}}^{opt}(\mathbf{y}) = \frac{E_x\{\mathbf{x} f_\mathbf{v}(\mathbf{y}-\mathbf{s}(\mathbf{x})\}}{E_x\{f_\mathbf{v}(\mathbf{y}-\mathbf{s}(\mathbf{x})\}}, \quad (8)$$

$$P^{opt}(\mathbf{y}) = \frac{E_x\{(\mathbf{x}-\hat{\mathbf{x}}^{opt}(\mathbf{y}))(\mathbf{x}-\hat{\mathbf{x}}^{opt}(\mathbf{y}))^T f_\mathbf{v}(\mathbf{y}-\mathbf{s}(\mathbf{x})\}}{E_x\{f_\mathbf{v}(\mathbf{y}-\mathbf{s}(\mathbf{x})\}}, \quad (9)$$

where $f_\mathbf{v}(\mathbf{y}-\mathbf{s}(\mathbf{x})) = \exp\left(-\frac{1}{2}(\mathbf{y}-\mathbf{s}(\mathbf{x}))^T \mathbf{P}_\mathbf{v}^{-1}(\mathbf{y}-\mathbf{s}(\mathbf{x}))\right)$.

Thus, in order for the optimal estimate to be calculated, it is necessary to calculate three sets of integrals

$$I_1^{MC1} = E_x\{\mathbf{x} f_\mathbf{v}(\mathbf{y}-\mathbf{s}(\mathbf{x})\},$$

$$I_2^{MC1} = E_x\{f_\mathbf{v}(\mathbf{y}-\mathbf{s}(\mathbf{x})\},$$

$$I_3^{MC1} = E_x\{(\mathbf{x}-\hat{\mathbf{x}}^{opt}(\mathbf{y}))(\mathbf{x}-\hat{\mathbf{x}}^{opt}(\mathbf{y}))^T f_\mathbf{v}(\mathbf{y}-\mathbf{s}(\mathbf{x})\},$$

representing the mathematical expectations corresponding to the function $f(\mathbf{x})$. In accordance with the Monte-Carlo method it is necessary to simulate a set of independent-of-each-other samples

$$\{\mathbf{x}^{(j)}\},\ j = \overline{1.n_1}, \quad (10)$$

with the p.d.f. $f(\mathbf{x}) = N(\mathbf{x};0,\mathbf{P}_0)$. Then according to (8) and (9), the nonlinear optimal estimate and its covariance matrix can be calculated as

$$\hat{\mathbf{x}}(\mathbf{y}) \approx \sum_{j=1}^{n_1^{MC1}} \mathbf{x}^j \mu^j, \quad (11)$$

$$P^{opt}(\mathbf{y}) \approx \sum_{j=1}^{n_1^{MC1}} (\mathbf{x}^j - \hat{\mathbf{x}}^{opt}(\mathbf{y}))(\mathbf{x}^j - \hat{\mathbf{x}}^{opt}(\mathbf{y}))^T \mu^j, \quad (12)$$

where $\mu^j = \tilde{\mu}^j / \sum_{j=1}^{n_1^{MC1}} \tilde{\mu}^j$ and

$$\tilde{\mu}^j = \exp\left(-\frac{1}{2}(\mathbf{y}-\mathbf{s}(\mathbf{x}^j))^T \mathbf{P}_\mathbf{v}^{-1}(\mathbf{y}-\mathbf{s}(\mathbf{x}^j))\right). \quad (13)$$

It is not difficult to understand that the size of the necessary calculations essentially depends on the number of integrals to be calculated and their complexity. The number of n-multiple integrals needed for the calculation of an estimate is equal to $n+1$, and for the calculation of the covariance matrix – $n(n+1)/2$. Therefore the total number of integrals is determined as

$$N_1 = (n+1)(1+n/2).$$

The integral complexity depends on the number of measurements.

3.2 Linear optimal estimate

Taking into account (1), it is easy to write

$$\overline{\mathbf{y}} = \int \mathbf{s}(\mathbf{x}) f(\mathbf{x}) d\mathbf{x} = E_x\{\mathbf{s}(\mathbf{x}) f(\mathbf{x})\}, \quad (14)$$

$$\mathbf{P}_{\mathbf{yy}} = E_x\{\mathbf{s}(\mathbf{x})\mathbf{s}(\mathbf{x})^{\mathrm{T}}\} - \overline{\mathbf{y}}\overline{\mathbf{y}}^{\mathrm{T}} + \mathbf{P}_\mathbf{v}, \quad (15)$$

$$\mathbf{P}_{\mathbf{xy}} = \int \mathbf{x}\mathbf{s}(\mathbf{x})^{\mathrm{T}} f(\mathbf{x}) d\mathbf{x} = E_x\{\mathbf{x}\mathbf{s}(\mathbf{x})^{\mathrm{T}}\}. \quad (16)$$

Using the Monte-Carlo method it is possible to write:

$$\overline{\mathbf{y}} \approx \frac{1}{n_1^{MC2}} \sum_{j=1}^{n_1^{MC2}} \mathbf{s}(\mathbf{x}^{(j)}), \quad (17)$$

$$\mathbf{P}_{\mathbf{yy}} \approx \frac{1}{n_1^{MC2}} \sum_{j=1}^{n_1^{MC2}} \mathbf{s}(\mathbf{x}^{(j)})\mathbf{s}(\mathbf{x}^{(j)})^T - \overline{\mathbf{y}}\overline{\mathbf{y}}^T + \mathbf{P}_\mathbf{v}, \quad (18)$$

$$\mathbf{P}_{\mathbf{xy}} \approx \frac{1}{n_1^{MC2}} \sum_{j=1}^{n_1^{MC2}} \mathbf{x}^{(j)}\mathbf{s}(\mathbf{x}^{(j)})^T. \quad (19)$$

Taking into account these moments, the linear optimal estimate and the corresponding covariance matrix can be calculated using (5)-(7). As follows from the relations given above, the number of n-multiple integrals needed for the calculation of the linear estimate is determined as

$$N_2 = m + m(m+1)/2 + nm = (m+1)(n+m/2).$$

In this case the complexity of the integral does not depend on the number of measurements.

4. NEURAL NETWORKS ALGORITHMS

The approximate solution of the estimation problem can be found by using a supervised NN (Haykin, 1994; Stepanov and Amosov, 2006), i.e.

$$\hat{\mathbf{x}}^{NN}(\mathbf{y}) = \mathbf{K}^{NN}(\mathbf{y}, \tilde{\mathbf{W}}), \quad (20)$$

where $\mathbf{K}^{NN}(\mathbf{y}, \widetilde{\mathbf{W}})$ is a special, in a general case, nonlinear function; $\widetilde{\mathbf{W}}$ is the matrix that specifies the free parameters (biases and weighting coefficients) and $\mathbf{y}$ is the input of the NN. The distinguishing feature of the supervised NN is the fact that it needs supervised learning, which allows finding the matrix $\widetilde{\mathbf{W}}$. Supervised learning calls for a training set and a cost function (learning criterion). Here the training set can be derived by simulation using the p.d.f $f(\mathbf{x})$, $f_{\mathbf{v}}(\bullet)$ and Eq. (1), i.e.

$$\{(\mathbf{y}^{(j)}, \mathbf{x}^{(j)})\}, \; j = \overline{1.n_2}, \qquad (21)$$

where the pairs $\mathbf{y}^{(j)}$, $\mathbf{x}^{(j)}, j = \overline{1.n_2}$ are the independent-of-each-other samples of the random vectors with the p.d.f. $f(\mathbf{x}, \mathbf{y})$. The learning criterion can be determined as

$$\widetilde{J}(\widetilde{\mathbf{W}}) = \frac{1}{n_2} \sum_{j=1}^{n_2} \left\| \mathbf{x}^{(j)} - \hat{\mathbf{x}}^{NN(j)}(\mathbf{y}^{(j)}, \widetilde{\mathbf{W}}) \right\|^2, \quad (22)$$

where $\hat{\mathbf{x}}^{NN(j)}(\mathbf{y}^{(j)}, \widetilde{\mathbf{W}})$ is the estimate generated by the NN using the measurements $\mathbf{y}^{(j)}$ corresponding to the realization of $\mathbf{x}^{(j)}$.

In this case the estimation problem can be formulated as follows: having the set (21) and the measurement $\mathbf{y}$, to find the estimate (20) that minimizes the criterion (22). It is evident that the criterion (22) tends to (2) as n_2 increases. So the estimate (20), optimal in the sense of the criterion (22), will be similar to the optimal Bayesian estimate (3). Thus, the problem of designing the estimation algorithm is reduced to selecting and learning of the NN, i.e. finding the parameters $\widetilde{\mathbf{W}}$ determined by the minimization of the criterion (22) formed with the use of the training set (21).

4.1 Linear optimal estimate

For determination of the linear optimal estimate it will be logical to use a linear NN. Taking into consideration the dimensions of the vector to be estimated, the linear NN $\hat{\mathbf{x}}^{NN}(\mathbf{y}, \widetilde{\mathbf{W}})$ can be written as follows:

$$\hat{\mathbf{x}}^{NN}(\mathbf{y}, \widetilde{\mathbf{W}}) = \mathbf{w}_0 + \mathbf{W}\mathbf{y}, \qquad (23)$$

where $\widetilde{\mathbf{W}} = [\mathbf{w}_0 \mid \mathbf{W}]$ is an $n \times (m+1)$-dimensional matrix that includes an n-dimensional bias vector $\mathbf{w}_0 = [w_{10} \; \ldots \; w_{n0}]^T$ and an $n \times m$-dimensional matrix of weighing coefficients $\mathbf{W} = [\mathbf{w}_1 \mid \ldots \mid \mathbf{w}_l \mid \ldots \mathbf{w}_n]^T$, in which $\mathbf{w}_l = [w_{l1} \; \ldots \; w_{lm}]^T$ are m-dimensional vectors $l = \overline{1.n}$. This NN has a single neuron layer. The number of neurons is the same as the dimensions of the estimated vector $\mathbf{x}$, and their activation function that depends on the scalar argument s represents an identical transformation (linear activation function), i.e. $\psi(s) = s$, $-\infty < s < \infty$. For the learning of the linear NN it is convenient to use the ordinary rule of Widrow–Hoff. It is not difficult to show that the estimate $\hat{\mathbf{x}}^{NN}(\mathbf{y}, \widetilde{\mathbf{W}})$ derived by the measurements $\mathbf{y}$ with the use of NN (23) trained in accordance with (22) can be represented as (Stepanov and Amosov, 2006):

$$\hat{\mathbf{x}}^{NN}(\mathbf{y}, \widetilde{\mathbf{W}}) = \overline{\mathbf{x}}^* + \mathbf{P}_{\mathbf{xy}}^* \left(\mathbf{P}_{\mathbf{yy}}^* \right)^{-1} [\mathbf{y} - \overline{\mathbf{y}}^*],$$

where $\overline{\mathbf{x}}^*$; $\overline{\mathbf{y}}^*$; $\mathbf{P}_{\mathbf{yy}}^*$, $\mathbf{P}_{\mathbf{xy}}^*$ are the sample values of the mathematical expectations and the corresponding covariance matrices, for example, $\overline{\mathbf{x}}^* = \frac{1}{n_2} \sum_{j=1}^{n_2} \mathbf{x}^{(j)}$.

It is clear that these procedures result in the estimate that is similar to the estimate obtained by the Monte-Carlo method. At the same time it should be noted that no matrix inversion is needed when using NN. This is due to the fact that learning results in the formation of $\mathbf{w}_0$, $\mathbf{W}$, which makes it possible to directly evaluate the estimate (23). This fact can be regarded as an advantage of NN-based algorithms. However it should be noted that the algorithm considered above does not provide a calculation of the error covariance matrix.

4.2. Nonlinear optimal estimate

To design a "good" estimation algorithm based on a nonlinear NN, it is necessary to choose a "good" NN and a "good" learning algorithm, i.e. an algorithm that provides such a $\widetilde{\mathbf{W}}$ that minimizes the learning criterion (22). Multi-layer feed-forward NN are used in this paper (Haykin, 1994). Further, in particular, a two-layer NN with feed-forward connections is used, with m-inputs, q-neurons in the hidden layer, and n- neurons in the output layer. In this case the estimates $\hat{x}_i^{NN}(\mathbf{y})$, $i = \overline{1.n}$ can be written as

$$\hat{x}_i^{NN}(\mathbf{y}) = \psi\left(\sum_{\mu=1}^{q} \left(w_{i\mu}^2 \varphi \left(\sum_{j=1}^{m} (w_{\mu j}^1 y_j) + w_{\mu 0}^1 \right) \right) + w_{i0}^2 \right), \qquad (24)$$

where $\varphi(s) = th\, s$, $\psi(s) = s$ are the activation functions for the neurons of the hidden and the output layers; ($w_{\mu 0}^1$, $w_{\mu j}^1$), (w_{i0}^2, $w_{i\mu}^2$), $\mu = \overline{1.q}$, $j = \overline{1.m}$, $i = \overline{1.n}$ are biases and weights of neurons of the hidden and the output layers. For the learning of the NN was used the back-propagation algorithm.

5. NONLINEAR NAVIGATION PROBLEM

Consider the estimation problem of an unknown vector $\mathbf{x} = (x_1, x_2)^T$ determining a vehicle's position on the plane with the use of the measurements of the ranges to the two reference beacons whose coordinates are assumed to be known. These measurements can be written as

$$y_i^k = s_i(\mathbf{x}) + v_i^k = \sqrt{\left(x_1 - x_1^i\right)^2 + \left(x_2 - x_2^i\right)^2} + v_i^k,$$
$$i = \overline{1.2}, \; k = 1,2...,$$

where $\mathbf{x}^i = (x_1^i, x_2^i)^T$, $i = \overline{1.2}$ are coordinates of the beacons; v_i^k are the measurement errors. Suppose that $\mathbf{x} = (x_1, x_2)^T$ is the zero mean Gaussian vector with a diagonal covariance matrix and similar variances σ_0^2; v_i^k are zero mean, independent-of-each-other and of $\mathbf{x}$, Gaussian random values with similar variances equal to r^2. Under the assumptions made $\mathbf{P}_0 = \sigma_0^2 \mathbf{E}_2$, $\mathbf{P}_\mathbf{v} = r^2 \mathbf{E}_{2i}$, where $\mathbf{E}_2$ and $\mathbf{E}_{2i}$ are 2×2 and $2i \times 2i$ unit matrices. It is also supposed that $\sigma_0 = 500\ m$; $r = 30\ m$; $\mathbf{x}^1 = (3000\ m, 0\ m)^T$; $\mathbf{x}^2 = (0\ m, 3000\ m)^T$.

Below are the simulation results obtained with the use of the following algorithms:
MC1 – the nonlinear optimal algorithms (11), (12) based on the Monte-Carlo method; MC2 – the linear optimal algorithms using the Monte-Carlo method for the calculation of (14)-(16); NN1 – the nonlinear optimal algorithm (24) using the two-layer nonlinear NN with m inputs, $q = 20$ neurons in the hidden layer and two neurons in the output layer; NN2 – the linear optimal algorithm using the single-layer linear NN (23) with m inputs and two output neurons.

The Cramer-Rao inequality was used to evaluate the potential accuracy. The latter makes it possible to find the lower Cramer-Rao boundary (CRB) for the unconditional covariance matrix of optimal estimate errors (Stepanov, 1998, Bergman 1999). It can be shown that for the problems considered the inequality can be written as

$$\mathbf{G}^{opt} \geq \mathbf{J}^{-1}, \qquad (25)$$

where

$$\mathbf{J}^{-1} = \left(\frac{1}{\sigma_0^2} \mathbf{E}_2 + \frac{1}{r^2} \int \left[\frac{d\mathbf{s}^T(\mathbf{x})}{d\mathbf{x}} \frac{d\mathbf{s}(\mathbf{x})}{d\mathbf{x}^T} \right] f(\mathbf{x}) d\mathbf{x} \right)^{-1}, \qquad (26)$$

$$\mathbf{G}^{opt} = \int \int (\mathbf{x} - \hat{\mathbf{x}}^{opt}(\mathbf{y}))(\mathbf{x} - \hat{\mathbf{x}}^{opt}(\mathbf{y}))^T f(\mathbf{x}, \mathbf{y}) d\mathbf{x} d\mathbf{y}, \qquad (27)$$

is the unconditional covariance matrix for the optimal estimation errors. It is easy to show that the square roots of the diagonal elements (26), determining the CRB, can be calculated as

$$\sigma_m^{CRB} \approx \left(\frac{1}{\sigma_0^2} + \frac{r^2}{m} \right)^{-1/2}.$$

The square roots of the diagonal elements (27) are the root-mean- square (RMS) errors for the corresponding algorithms. They characterize the accuracy of algorithms needed for comparison. The diagonal elements (27) were calculated as

$$\mathbf{G}_{\mu\mu}^s \approx \frac{1}{L} \sum_{j=1}^{L} ((\mathbf{x}_\mu^j - \hat{\mathbf{x}}_\mu^s(\mathbf{y}^j))^2, \qquad (28)$$
$$\mu = 1, 2; \; s = MC1, MC2, NN1, NN2.$$

Figure 1 presents σ_m^{CRB} and σ_m^{MC1}, σ_m^{MC2}, σ_m^{NN1}, σ_m^{NN2} which are the CRB and RMS errors at a different number of measurements $m = 2k$, $k = 1,2...$ corresponding to the Monte-Carlo and NN methods for the linear and nonlinear estimates.

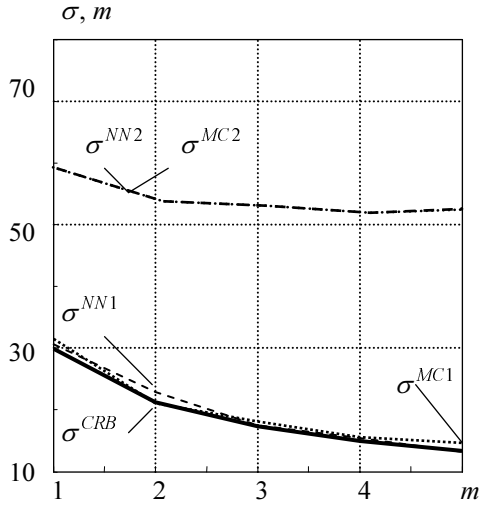

Fig. 1. RMS errors.

The simulation results are shown for one of the component coordinates. They look similar for the other component as well. For the sake of simplicity the indices m are not shown in the plots. The numbers of the samples n_1^{MC1} and n_1^{MC2} for the nonlinear and linear optimal algorithms based on the Monte-Carlo method are equal to 30000 and 3000. The number of the samples for learning of the nonlinear n_2^{NN1} and linear n_2^{NN2} algorithms based on the NN is equal to 3000. The number of the samples for testing is L=300. The analysis of the plots allows making the following conclusions.

As one would expect, the RMS errors for linear and nonlinear optimal estimates calculated both with the use of the Monte-Carlo method and NN are the same. This is due to the fact that the numbers n_1^{MC1},

n_1^{MC2}, n_2^{NN1} and n_2^{NN2} were chosen such that calculation errors in each of these methods would not have affected the result.

The RMS errors for the linear optimal estimates are different from the CRB because of the errors caused by the linear character of the algorithms. The RMS errors for the nonlinear estimates do not practically differ from the CRB. The number of the samples n_1^{2} that provides a low level of calculation errors in the Monte-Carlo method at the realization of linear estimates is close to the number of the samples n_2^{NN2} used in NN learning.

At the calculation of nonlinear estimates the number n_1^{1} substantially exceeds n_2^{NN1}, which, in its turn, differs little from n_2^{NN2}.

6. CONCLUSIONS

It is possible to outline two variants of applications of supervised NN for nonlinear estimation problems. The first line presumes the availability of a training set from experimental data. In this case it is possible to speak about potential adaptive properties of the algorithms designed. This situation is not specific for some applied problems.

For the second line NN is employed as a numerical procedure for realization of traditional (Bayesian) algorithms. The training set is formed by simulation using prescribed models. In this case there is no question of any adaptive properties due to NN. The goal is to develop economical, from the computational standpoint, procedures, using NN for calculation of optimal estimates. The paper has considered the peculiarities of using NN in estimation problems exactly for this case in comparison with the Monter-Carlo method. The results of the studies allow pointing out the following.

In designing linear estimates the problems of choosing the NN itself and its learning are easy to solve. It is explained by the fact that it is possible to use the linear NN which is easy to learn using the rule of Widrow–Hoff. The numbers n_1^{MC2} and n_2^{NN2} are similar, thus the size of calculations needed for the Monte-Carlo method and for NN learning is also comparable in this case.

In searching for nonlinear estimates there arises the problem of choosing NN and the algorithm for its learning as the size of calculations essentially depends on the type of NN. In the realization of the nonlinear algorithm the number n_1^{MC2} in the example considered exceeds n_2^{NN2} by an order. The question of relation between these numbers needs additional study with consideration of possible reduction of the size of calculations with the use of various modifications of the Monte-Carlo method, particle and unscented filters and so on (Daum, 2005).

In calculating estimates with the use of the neural network after its learning, the algorithm for finding estimates becomes trivial and is easier to realize than with the use of the Monte-Carlo method.

An essential disadvantage of NN algorithms is the fact that their designing does not presume the calculation of the error covariance matrix. In a number of applied problems, particularly, in navigational problems such characteristic is of great importance. This means that one more NN is required for deriving it.

REFERENCES

Bergman, N. (1999). Recursive Bayesian estimation. Navigation and Tracking Applications. *Linkoping Studies in Science and Technology. Dissertations, 579,* Sweden.

Chaffee, J. and K. Kovach (1996). Bancroft's Algorithm is Global Nonlinear Least Squares. *Proc. of ION GPS-96, The Institute of Navigation, Kansas City, Missouri,* September 17–20.

Daum, F. (2005). Nonlinear Filters: Beyond the Kalman Filter. *IEEE Aerospace and Electronic Systems. Tutorials,* 20(8), pp. 57-71.

Haykin, S. (1994). *Neural networks: A comprehensive foundation.* MacMillan College, New York.

Haykin, S. (2001). *Kalman Filtering and Neural Networks.* MacMillan College, New York.

Meditch, J.S. (1969). *Stochastic optimal linear estimation and control.* Mc. Graw Hill, New York.

Jazwinski, A.H. (1970). Stochastic processes and filtering theory. In: *Academic Press, New York.*

Stepanov, O.A. (1998). Nonlinear filtering and its application in navigation. *St. Petersburg, Russia, State Research Center of Russia - Central Scientific & Research Institute Elektropribor,* pp. 343. In Russian.

Stepanov, O.A. and O.S. Amosov (2006). Optimal Estimation Algorithms Based on the Monte Carlo Method and Neural Networks for Nonlinear Navigational Problems. In: *Proc. of the CAC/CACD/ICC,* Munich, Germany.

Zaritsky, V.S., V.B. Svetnik and L.I. Shimelevich (1975). The Monte-Carlo techniques in problems of optimal information processing. *Automation and Remote* Control, 36, pp. 2015–2022.