http://ansinet.com/itj

ITJ

ISSN 1812-5638

# INFORMATION TECHNOLOGY JOURNAL

ANSInet
Asian Network for Scientific Information
308 Lasani Town, Sargodha Road, Faisalabad - Pakistan

Information Technology Journal 12 (24): 7983-7988, 2013
ISSN 1812-5638 / DOI: 10.3923/itj.2013.7983.7988


# Application of Neural Network Filter to Estimate the Velocity of Targets in Strong Clutter

Zhang xiaoyun
Department of Information Engineering,
Chongqing University of Science and Technology, Chongqing, China

**Abstract:** The velocity estimation is an important problem in many applications such as communication and navigation. But in some application such as sea, this task is very difficult because of the strong clutter. Many algorithms have been proposed for this problem. The Maximum Likelihood (ML) is one of the good solutions. This paper describes an application of Neural Network (NN) for obtaining the global optimal solution of ML velocity estimation. It overcomes the local optima problem existing in some ML velocity estimation algorithms and improves the estimation accuracy. The computation complexity is modest.

**Key words:** Artificial neural network, velocity estimation, maximum likelihood, target detection, parameter estimation

## INTRODUCTION

Motivated by the MIMO technique in communication systems, the two new concepts of MIMO radar are introduced. One is transmitting diversity MIMO radar, the other one is receiving and transmitting diversity MIMO radar. The proposed MIMO radar enjoys the same benefits that MIMO communication systems have. Specifically, the transmitting diversity MIMO radar can greatly improve the radar's performance over traditional radar on anti-intercept of radar signal, weak target detection, etc.

In many applications such as radar and navigation, the estimation of target velocity is one of the important problems. Lots of techniques for this problem have been proposed over past decades. The Maximum Likelihood technique is one of the first to be investigated and best in theory. Nonetheless, because of the high computational load of the multivariate nonlinear maximization problem involved, it does not become popular. Instead, suboptimal method with reduced computational load have governed the field. The better known ones are the MUSIC method of Schmidt (Schmidt, 1986) and the minimum norm method of Reddi (Reddi, 1979) and Kumaresan and Tufts (Sharman, 1988).

However, the ML method over-performs other methods in many aspects (Schweppe, 1968), especially, when the target echo is very small or when the noise or clutter is very strong. In fact, many techniques cannot deal with the circumstances of coherent signals.

Many researchers have proposed various algorithms to maximize the likelihood function, wanting to guarantee global convergence within less computing time. Alternating projection method (Ziskind and Wax, 1988), simulated annealing algorithm (Godara, 1997), grid search approach, data-supported grid search (Stoica and Gershaman, 1999), can approximately obtain the ML estimation. But most of them cannot guarantee global convergence in general case(Stoica and Sharman, 1990).

In this study, a global optimization of neural network (Kumaresan and Tufts, 1983) is developed to search for the nonlinear global optimization solution of the maximum likelihood in radar application. The target velocity is estimated from the received signal of the array (Shi and Eberhart, 1998). And then, we study the performance of AN algorithm.

## SIGNAL MODEL

Consider an array constituted of M sensors with arbitrary locations and arbitrary directional characteristics and assume that L narrow-band plane waves arriving on the array from locations $\theta_1$, $\theta_2$,..., $\theta_L$ and the velocity of the target is $v_1$, $v_2$,..., $v_L$ according to the s.

Since narrow-band in the sensor array context means that the propagation delays of the signals across the array are much smaller than the reciprocal of the bandwidth of the signals, it follows that the complex envelopes of the signals received by the array can be expressed as:

$$X(t)=\sum_{k=1}^{L} a(\theta_k)s_k(t)\cdot e^{j2\pi f v_k}+n(t) \tag{1}$$

where, x(t) is the $M\times 1$ vector:

$$X(t) = [x_1(t),\ x_2(t),..., x_M(t)]^T \quad (2)$$

where, T is the transpose. And $a(\theta_k)$ is the steering vector of the array toward direction $\theta_k$.

$$a(\theta_k) = [a_1(\theta_k)e^{-j\omega_0\tau_1(\theta_k)}, \ldots, a_M(\theta_k)e^{-j\omega_0\tau_M(\theta_k)}]^T \quad (3)$$

$$n(t) = [n_1(t), n_2(t),..., n_M(t)]^T \quad (4)$$

Here:

$x_i(t)$ = The signal received by the ith sensor

$S_k(t)$ = The signal emitted by the kth source

$a_i(\theta_k)$ = The amplitude response of the ith sensor to a wavefront impinging from location $\theta_h$

$n_i(t)$ = The noise at the ith sensor

Then, consider of the clutter signal, it is similar to target echo. But it is a random signal which can be described by a coefficient $\zeta_{JK}$.

$$C(t)=\sum_{k=1}^{L}\zeta_{Jk}a(\theta_k)s \quad (5)$$

So, the signal received can be expressed as:

$$X(t)=\sum_{k=1}^{L}a(\theta_k)s_k(t)e^{j2\pi fv}+C(t)+n(t) \quad (6)$$

The vector of the received signals x(t) can be expressed more compactly as:

$$x(t) = A(\Theta s(t)+n(t) \quad (7)$$

where, the $A(\Theta)$ is the M×L matrix of the steering vectors $A(\Theta) = [a(\theta_1,..., a(\theta_L)]$. And s(t) is L×1 vector of the signals $s(t) = [s_1(t),..., s_L(t)]$.

The localization problem is to estimate the locations $\theta_1, \theta_2,..., \theta_L$ of the sources from N samples ("snapshots") of the received signals. The maximum likelihood estimation of the source localization problem is derived as:

$$\hat{\Theta} = \arg\{\max_T L(T)\} = \arg\{\max_T tr(P_{A(T)}R)\} \quad (8)$$

where, tr[ ] is the trace of the bracketed matrix, $P_{A(\Theta)} = A(\Theta)(A^H(\Theta)A(\Theta))^{-1}A^H$ is the projection operator onto the space spanned by the columns of the matrix $A(\Theta)$:

$$R=\frac{1}{N}\sum_{i=1}^{N}X(t_i)X^H(t_i)$$

is the sample covariance matrix and H denotes the Hermitian conjugate. In this paper, we use the proposed PSO algorithm [11] as the optimization tool, searching for the global optimal solution.

## NEURAL NETWORK ALGORITHMS

The theory of linear optimum filters is based on the mean-square error criterion. The Wiener filter that results from the minimization of such a criterion and which represents the goal of linear adaptive filtering for a stationary environment, can only relate to second-order statistics of the input data and no higher.

This constraint limits the ability of a linear adaptive filter to extract information from input data that are non-Gaussian. The use of a Wiener filter or a linear adaptive filter to extract signals of interest in the presence of such non-Gaussian processes will therefore yield suboptimal solutions. Despite its theoretical importance, the existence of Gaussian noise is open to question. Although, by so doing, we no longer have the Wiener filter as a frame of reference and so complicate the mathematical analysis, we would expect to benefit in two significant ways: improving learning efficiency and a broadening of application areas.Moreover, non-Gaussian processes are quite common in many signal processing applications encountered in practice. We may overcome this limitation by incorporating some form of nonlinearity in the structure of the adaptive filter to take care of higher order statistics.

This terminology is derived from analogy with biological neural networks that make up the human brain.In this section, we describe an important class of the nonlinear adaptive system commonly known as artificial neural networks or just simply neural networks.

A neural network is a massively parallel distributed processor that has a natural propensity for storing experiential knowledge and making it available for use. The neural network filter consists of a feed-forward neural network with two layers.

The Hopfield model neural network is a single layer of fully inter-connected neurons that update their outputs upon sampling the outputs of other neurons in the network, via the synaptic link.

The synaptic link between the ith and the jth neurons, in a network of P neurons, form a symmetric matrix T which elements obey follow formula:

$$t_{ij} = -t_{ij}; \quad t_{ii} = 1 \tag{9}$$

The network changes state using the following dynamic equation:

$$C_i \frac{du_i}{dt} = \sum_j^K k_{ij} w_j + N_i \tag{10}$$

where, Matrix C is the input capacitance of the ith neuron, Ii is the external input and is the internal state of the neuron. The output state vi of the neuron is given by the following nonlinear transformation:

$$v_i = g_i = \frac{1}{1 + e^{-\frac{v_i}{\eta}}} \quad \text{for} \quad v_i \in \{-1, +1\} \tag{11}$$

where, $g_i$ is the sigmoid transfer function of the ith neuron and $1/\eta$ is the gain of the neuron. The network dynamic equation defines a complex system but it is possible to find an energy function satisfying the Liapunov's stability criterion:

$$E = -\frac{1}{2} \sum \sum w_{ij} v_i v_j - \sum v_i I_i \tag{12}$$

The back-propagation algorithm has emerged as the workhorse for the design of a special class of layered feed-forward networks known as multilayer perceptions. Input layer of nodes, which provide the means for connecting the neural network to the source(s) of signals driving the network. Output layer of processing units, which provide one final stage of computation and thereby produce the response of the network to the signals applied to the input layer.One or more hidden layers of processing units, which act as "feature detectors".

The processing units are commonly referred to as artificial neurons or just neurons. Typically, a neuron consists of a linear combiner with a set of adjustable synaptic weights, followed by a nonlinear activation function; two commonly used forms of the activation function $\phi\{\bullet\}$ are shown in Fig. 1.

The first one, shown in Fig. 1, is called the hyperbolic function, the decision space of a P-neuron network is represented by a P-dimensional hypercube $D([0,1]^P)$. The network starts from some initial state within $D([0,1]^P)$ and developed towards one of the corners that corresponds to minimum. Each corner of this hypercube represents a possible digit output state of the network; one of these corners represents the solution state and one or more of the other corners represents the local-minima of the energy function.

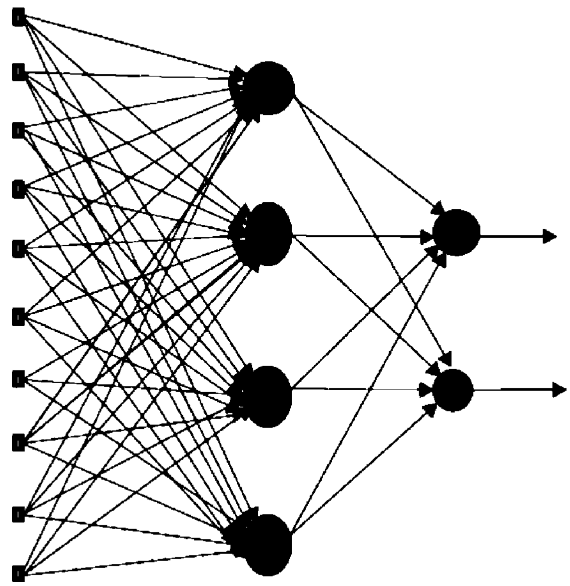

Fig. 1: Fully connected feed-forward of acyclic network

For a digit neuronal trader function, the global minimum must be one of the comers of the hypercube. Yet provided that the neuronal gain is high enough to eliminate the perturbation of the digit energy function, the analog network can be replaced by digital network.

However for an analog network with a sigmoid transfer function, due to the reason of an analog transfer function represent a perturbation to the energy function; the global minimum must not be one corner of the hypercube.

Therefore the global minimum may be thought to be one of the corners of hypercube. Then after the computation of the energy of all formula, the P is the global minimum which we need. We can calculate the energy of all m e m on the hypercube directly so long as the P is not very large.

Here we call it Energy Comparing method (ECM).The advantage of ECM that any other modification doesn't pose is that it ultimate the local minimum completely. And the computational complexity does not increase obviously so long as the number of neurons is not very large.

For the pulse radar, first supposed that the target moves so slowly that the return sequences of the target in the H times Pulse repetition interval PRI are at the same position.

The changing pattern of the return sequences of the target will manifest some specific distributions in the amplitude and frequency feature spaces. Forth more, the distribution will possess considerable stability in a distance range.

Supposed is the column vector composed of the $t_{th}$ sampling point in the each return sequence. The effectiveness of feature extraction is guaranteed when the network is solved with the ECM. Of course, the best way of coding is that the distinct feature will be shown clearly after coding.Since the output of the network is zero and one, the features may be coded by the natural order of the output of the Hopfield networks.

When we have not any priori knowledge about the feature distributions, we may use the above coding method. The course of synthesizing two kinds of distinct features is, in fact, a course of features integration which transforms distinct input feature spaces into the same output space so as to make detection and decision.

Consider the simple case of a continuous function mapping from a 2-dimensional z, y input space to a 1-dimensional z output space. It is theoretically possible to model this mapping with a number of 2-dimensional radial functions. Our networks use Gaussian radial functions which, in 2-dimensions, look like bumps or hills. A radial function is one whose evaluation depends upon a radial distance from the function center.

RBF networks use memory-based learning for their design. Specifically, learning is viewed as a curve-fitting problem in high-dimensional space.Another popular layered feed-forward network is the Radial-basis Function (RBF) network, whose structure is shown in Fig. 2.

Learning is equivalent to finding a surface in a multidimensional space that provides a best fit to the training data 2. A commonly used formulation of the RBFs, which constitute the hidden layer, is based on the Gaussian function. Generalization (i.e., response of the network to input data not seen before) is equivalent to the use of this multidimensional surface to interpolate the test data.

To be specific, let u denotes the signal vector applied to the input layer and $u_i$ denote the center of the Gaussian function assigned to hidden unit i.

A velocity estimation network consists of three layers of nodes, input, Gaussian and output, which are fully-connected by two layers of arcs, center and weight. "Fully-connected" means all nodes in layer i are connected to all nodes in layer i+l. The function space is primarily shaped by the Gaussian nodes. There are no connections between nodes in the same layer.

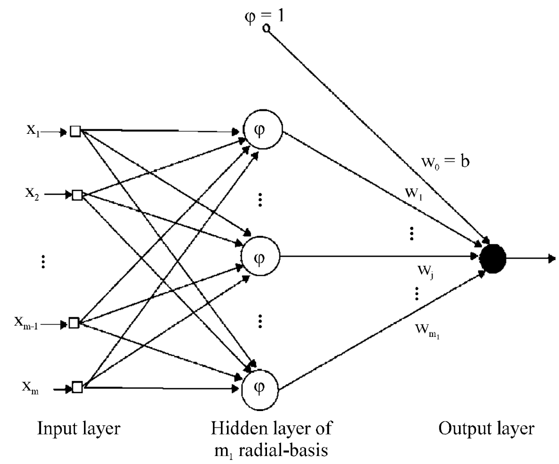


Fig. 2: RBF network

The input nodes, I, in the velocity estimation network accept preprocessed input data and fan it out to the center arcs. The center arcs connect each input to each Gaussian node and calculate the distance for that dimension from the current input value to a given Gaussian, k, i.e.:

$$C_{ij} = I_j = m_{jk} \tag{13}$$

The use of a linear output layer in an RBF network may be justified in light of Cover's theorem on the reparability of patterns. According to this theorem, provided that the transformation from the input space to the feature (hidden) space is nonlinear and the dimensionality of the feature space is high compared to that of the input (data) space, then there is a high likelihood that a no separable pattern classification task in the input space is transformed into a linearly separable one in the feature space or zero, that Gaussian is "relevant" to the answer, i.e. it's center in the function space is near the current input. Gaussian node H computes the following exponential.

## SIMULATION

In order to demonstrate the performance of the ML estimator computed by our proposed NN algorithm, some simulation is used .

In the experiments, the array is linear and uniform with three isotropic sensors spaced a wavelength apart. The sources are two equal power narrow-band emitters and the noise is additive and uncorrelated from sensor to sensor and with the signals.

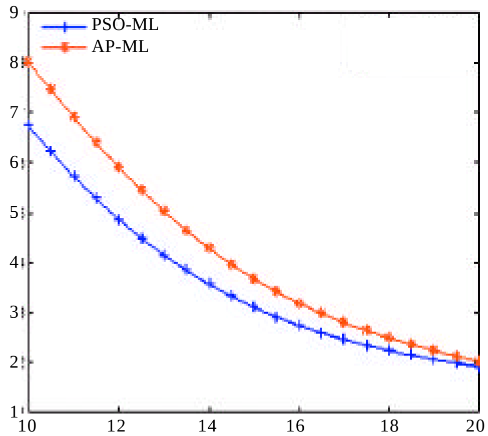


Fig. 3: Two equal power uncorrelated emitter, $v_1$ = 100 msec$^{-1}$. $v_2$ = 100 msec$^{-1}$, the x-axis denotes SNR, y-axis denotes RMS errors

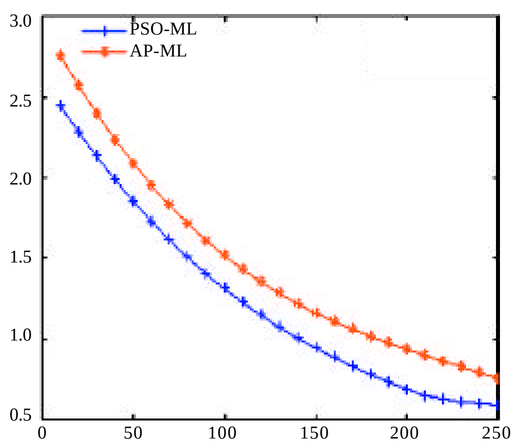


Fig. 4: Same scenario as in Fig. 5. The SNR is 20 dB, the x-axis denotes snapshots, y-axis denotes RMS erros

In every experiment we perform 5000 Monte-Carlo runs anad compute the root-mean-square (RMS) error for each velocity value. In all the experiments, the layer of neural network size is 40. All initial estimations are taken from the interval:

$$[-\frac{p}{2},\frac{p}{2}]$$

randomly. And the iteration time is 300 for the experiment.

In the first experiment we simulate two target with velocity of $v_1$ = 100 msec$^{-1}$. $v_2$ = 100 msec$^{-1}$ the number of snapshots taken is 10. Figure 3 shows the resulted rms error (in degrees) of the first source as a function of the SNR, defined as SNR = 10log($s^2/\sigma^2$) (where $s^2$ and $\sigma^2$ are the average power of the signals and the noise, respectively). The improved performance of the NN based ML estimator at low and moderate SNR is evident.

In the second experiment, the scenario is the same as in the first one, except that this time we fix the SNR to 20 dB. Figure 4 shows the resulted rms error of the first source as a function of the number of snapshots.

From the experiment, we can see NN algorithm outperforms AP algorithm. Moreover, in our experiment, AP algorithm sometimes does converge to a local optimum. And we solved it successfully by NN algorithms.

## CONCLUSION

We have proposed a new algorithm for computing the ML estimator of the direction of multiples source in the far field. The algorithm is iterative. The convergence of the NN algorithm to the global maximum is verified. The initial guess of direction does not influence the algorithm's convergence. This is conspicuous advantage over traditional algorithm. The complexity involved in each iteration is modest.

## ACKNOWLEDGMENTS

This study is supported by the National Natural Science Foundation of China under Grant No.60701015, the Natural Science Foundation of Chongqing under Grant No.20232004150020 and the Specialized Research Fund for Chongqing University of science and technology under Grant No.20110023110002.

## REFERENCES

Godara, L.C., 1997. Application of antenna arrays to mobile communications. II. Beam-forming and direction-of-arrival considerations. IEEE Proc., 85: 1195-1245.

Kumaresan, R. and D.W. Tufts, 1983. Estimating the angles of arrival of multiple plane waves. IEEE Trans. Aerosp. Electron. Syst., AES-19: 134-139.

Reddi, S.S., 1979. Multiple source location: A digital approach. IEEE Trans. Aerosp. Electron. Syst., AES-15: 95-105.

Schmidt, R.O., 1986. Multiple emitter location and signal parameter estimation. IEEE Trans. Antennas Propagat., 34: 276-280.

Schweppe, F.C., 1968. Sensor-Array data processing for multiple-signal sources. IEEE Trans. Inform. Theory, 14: 294-305.

Sharman, K.C., 1988. Maximum likelihood parameter estimation by simulated annealing. Proceedings of the International Conference on Acoustics, Speech and Signal Processing, April 11-14, 1988, New York, USA., pp: 2741-2744.

Shi, Y. and R. Eberhart, 1998. A modified particle swarm optimizer. Proceedings of the World Congress on Computational Intelligence and IEEE International Conference on Evolutionary Computation, May 4-9, 1998, Anchorage, AK., pp: 69-73.

Stoica, P. and A.B. Gershaman, 1999. Maximum-Likelihood DOA estimation by data-supported grid search. IEEE Signal Process. Lett., 6: 273-275.

Stoica, P. and K.C. Sharman, 1990. Maximum likelihood methods for direction-of-arrival estimation. IEEE Trans. Acoust. Speech Signal Process., 38: 1132-1143.

Ziskind, I. and M. Wax, 1988. Maximum likelihood localization of multiple sources by alternating projection. IEEE Trans. Acoust. Speech Signal Process., 36: 1553-1560.