Maximum Likelihood Estimation of Exponential Signals Using Artificial Neural Networks

Ching-Hui J. Ying, Stanley C. Ahalt, and Randolph L. Moses

Department of Electrical Engineering
The Ohio State University
2015 Neil Avenue
Columbus, OH 43210

Abstract

We present a neural network architecture for estimating the model parameters of noisy, superimposed signals. A maximum likelihood estimator using artificial neural networks is proposed. The well-known back propagation learning rule is used to modify the network weights and biases. We show that the resulting weights are the desired parameter estimates. To improve convergence initial estimates for the networks are obtained from FFT peaks. The advantages and applications of the proposed neural network maximum likelihood estimator are explored. Computer simulations are provided to demonstrate the performance of the proposed algorithm.

I. Introduction

The parameter estimation of superimposed exponential signals has been a topic of considerable attention, as the problem has applications in a number of areas, including speech processing, radar and sonar signal processing, and array processing. For the problem of estimating the model parameters, many algorithms have been developed, including iterative maximum likelihood (ML) methods [1, 2]. However, these iterative ML algorithms are computationally intensive, and can converge to false local minima if they are not properly initialized.

Neural networks has been an active area in the past few years, and researchers have applied neural networks to many applications. However, little work has been done with respect to the problem of estimating model parameters of noisy exponential signals. Recently Luo and Bao [3] developed a neural network which computes the projection operator involved in the ML estimation of the model parameters. Nevertheless, they do not consider the estimation problem as a whole, which includes model order determination and parameter estimation.

In this paper we propose a model order revealing ML estimator using an artificial neural network (ANN). The ML estimation is readily implemented in a neural architecture with local gradient and parallel computation. The parallel structure of the ANN effectively reduces the computational burden involved in the ML estimation. In the proposed algorithm the back propagation learning rule is used to train the ANN. Two architectures are presented, a compact architecture in which batch mode learning is needed, and an architecture in which batch mode learning is not required.

The neural-based ML estimator we propose is initialized and updated in such a way as to reduce problems associated with convergence to false local minima. The initialization is based on peak selection from an FFT of the signal; superresolution is achieved by splitting the FFT peaks using a criterion based on the Cramér-Rao bound (CRB). In addition, an outer learning loop is used to adaptively adjust the estimated model order on-line; this procedure provides some additional robustness with respect to initial conditions.

This paper is structured as follows. Section II presents the data model and the problem formulation. In Section III we discuss ML estimation and ANNs. In Section IV we present the proposed ML algorithm and an initialization procedure. Section V presents simulation studies which demonstrate the utility of the proposed algorithm. Finally, Section VI presents conclusions and ongoing work.

II. Data Model and Problem Formulation

Assume a data vector y of length m is modeled as a noisy undamped exponential sequence

$$y_q = x_q + e_q = \sum_{i=1}^{n} \alpha_i e^{j(\omega_i q+\phi_i)} + e_q \qquad q = 0, 1, \ldots, m-1, \tag{1}$$

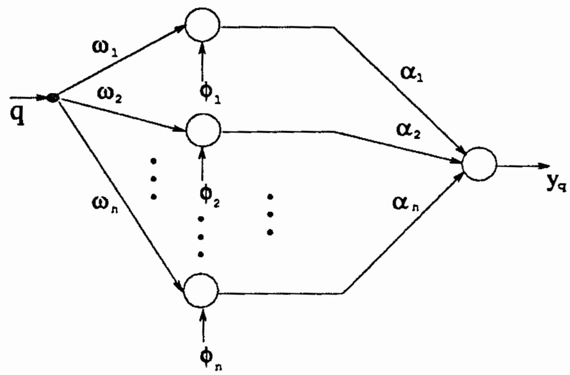

Figure 1: Maximum likelihood estimation using an artificial neural network.

where $\alpha_i \in R$, ω_i, $\phi_i \in (-\pi,\ \pi]$. It is assumed that the n exponential modes in the data are distinct *i.e.*, $\omega_k \neq \omega_l$ for $k \neq l$. It is also assumed that $\{e_q\}$ is a zero mean complex white Gaussian noise sequence with variance σ^2 and uncorrelated with the true data $\{x_q\}$.

Now define the parameter vector as $\theta^T = [\alpha_i, \phi_i, \omega_i]_{i=1}^n$. Thus, the parameter estimation of the undamped exponential signals can be formulated as: Given the data $\{y_q\}$, estimate the parameter vector θ.

III. Maximum Likelihood Criterion and Artificial Neural Networks

Under these assumptions made, the maximum likelihood estimation of the parameters of the undamped exponential model in Equation (1) is the same as minimizing the following loss function [1, 2]

$$E = \sum_{q=0}^{m-1} |y_q - x_q|^2, \tag{2}$$

where $|\cdot|$ is the magnitude of a complex number. The ML estimates of the parameters are obtained as the values which minimize E.

Consider the artificial neural network shown in Figure 1. In order to model the undamped exponential signals, let the weights and biases of the nodes be the frequencies, phases, and amplitudes of the signals as shown in the figure. Note that the input is the index of the reference (desired) data sequence, and activation functions of the hidden layer and the output layer are the complex exponential function and the simple linear function, respectively. If the ML loss function in Equation (2) and the batch mode backprop learning rule are used to train the ANN, the weights and biases are the ML estimates. Instead of employing batch mode backprop, we can also parallel m ANNs and calculate the gradients for each data point in parallel. This would reduce the computation by a factor of m at the expense of more hardware.

The parameter gradients needed for learning can be derived in a straightforward manner from equation (2) (the details are omitted for brevity). The updates of the ANN can be derived based on the gradients. Here we use the Quickprop algorithm presented in [4] to calculate the updated parameters. The advantage of the Quickprop algorithm is that when a large update which results in a minimum being passed ove is made, the next update will result in a point between the previous two points. Therefore, it minimizes oscillating phenomenon and thus accelerates convergence.

IV. Initialization Procedure and A Maximum Likelihood Algorithm

The iterative minimization performed by back propagation must be initialized. For this problem, we initialize the network weights and biases by use of an FFT. We take the m points FFT of $\{y_q\}_0^{m-1}$ and locate peaks (one can zero-pad the signal and then take the FFT). Because the FFT is unable to resolve closely spaced frequency components, we split each FFT peak into two or more initial frequency estimates using a criterion based on the CRB. The CRB gives a measure of how many peaks can be resolved as a function of SNR [5, 6]. We note that this "split" procedure often tends to estimate an excessively high model order which might normally be avoided because of high computation cost. However, with the neural