A Method of Genetic Algorithm (GA) for FIR Filter Construction: Design and Development with Newer Approaches in Neural Network Platform

Ajoy Kumar Dey, Susmita Saha
Department of Signal Processing
Blekinge Tekniska Hogskola (BTH)
Karlskrona, Sweden
e-mail: dey.ajoykumar@yahoo.com

Avijit Saha, Shibani Ghosh
Department of Electrical and Electronic Engineering
Bangladesh University of Engineering and Technology
Dhaka, Bangladesh
e-mail: avijit003@gmail.com

Abstract—**The main focus of this paper is to describe a developed and dynamic method of designing finite impulse response filters with automatic, rapid and less computational complexity by an efficient Genetic approach. To obtain such efficiency, specific filter coefficient coding scheme has been studied and implemented. The algorithm generates a population of genomes that represents the filter coefficient where new genomes are generated by crossover, mutation operations methods. Our proposed genetic technique has able to give better result compare to other method.**

Keywords-Genetic Algorithm; FIR: filter design; optimization; neural network.

I. INTRODUCTION

This paper represent a developed and dynamic method for genetic algorithm for to design a FIR filter in Neural network platform where the FIR filter has certain kinds of finite impulse response and genetic algorithm provide a automatic, efficient and less complex design method [6].

Select the standard polynomial transfer function by satisfied the response specification, followed by the implementation of the transfer function in one of the standard circuit structures is the conventional approach to filter design where optimization approach is required in many case of this approach [7]. The non standard response specification, the computation complexity for digital filters and many other ways we need to use this approach [3], [4].

We know that Genetic algorithm can be successfully employed for minimizing or maximizing a cost function based on the genetic and natural selection. The design of quantized digital FIR filters, being an optimization problem over a discrete coefficient space, can therefore be faced using a genetic approach [3]. Genetic algorithm optimization methods have emerged as a powerful approach to solving the more difficult optimization problems. This paper gives the attention to theoretical analysis of genetic algorithm and a efficient and developed method analysis for genetic algorithm for FIR filter design [1].

The paper organization is as follows: Section II describes the Genetic algorithm and its implementation. Section III discusses Research rationale. Section IV describes about the design method of FIR filter and Section V discusses about the Analysis of findings. Finally, Section VI presents a conclusion and an indication towards the future scope of this work.

II. GENETIC ALGORITHM (GA)

A. General Idea of GA

Genetic algorithms are stochastic strategies for optimization. Evolutionary computation is used by the genetic algorithms which one is one of the major features. As an optimization tool, it can be used as traning algorithm for any supervised Neural Network [5].

Natural selection and Natural genetics mechanics approaches first time introduced by the genetic algorithm where this two mechanism is the based part for the GA. Genetic algorithms may be differentiated from more conventional techniques by these characteristics, a) direct manipulation of the encoded representation of variables at the string level, rather than manipulation of the variables themselves, b) search form a population of points rather than form a single point, thus reducing the probability of reaching a false peak, c) blind search by sampling, ignoring all information except the outcome of the sample, d) use of stochastic rather than deterministic operators [6].

Effectiveness in searching large, noisy, multimodal problem spaces and versatility of genetic algorithm gave by these characteristics where smooth and differentiable surfaces are to be searched, calculus based methods are likely to do better and where the problem space is small, genetic algorithms may show no advantage over enumerative and random search methods .
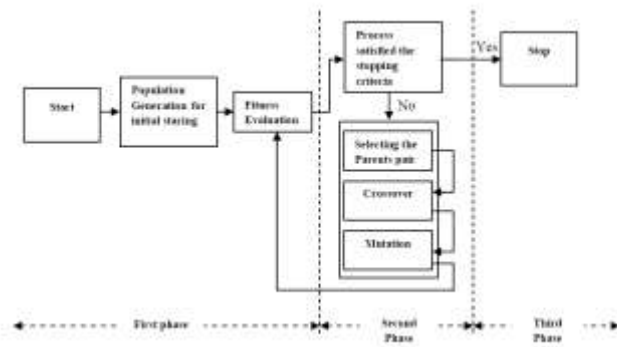
Figure 1.   Natural Flow of Genetic algorithm.

Inherently parallel search performed by the genetic algorithms and make effective use of historical information in their search for improved areas of performance as well [2]. They are not the only class of optimization technique to have been inspired by evolutionary process where evolutionary strategies based on selection according to fitness, mutation and reproduction.

There are many advantages and drawback in genetic algorithm. i) It can escape local minimums in the error function, ii) any error function can be used easily (not only the least square error function), iii) genetic algorithm can be easily implemented in parallel computers; those are consider

as the benefits with genetic algorithm. As well so many disadvantages with genetic algorithm like, a) the

computational cost is high, b) the convergence speed varies strongly with different tasks, c) some optimization problems are not solvable with genetic algorithms.

The genetic algorithm described in this paper in a simple way only the process of selection according to fitness, crossover, mutation and reproduction.

B.   Structure of Genetic Algorithm

There are many ways to implements a GA but most of genetic algorithm consists of an iteration of 3 steps (from figure 1). Like, 1) selection of parent (we do not really need two parents even though it is normally used. In real life the use of two parents is one way to make the iteration process continue.), 2) Creation of new population, 3) Mutation of the population.

Now we consider the multilayer feed forward neural network to use genetic algorithm in order to train the network. Assume we have a training sequence, $(x_1, t_1), (x_2, t_2), \ldots, (x_N, t_N)$ and we have initialized the weights W_1, W_2, W_3 (parents)

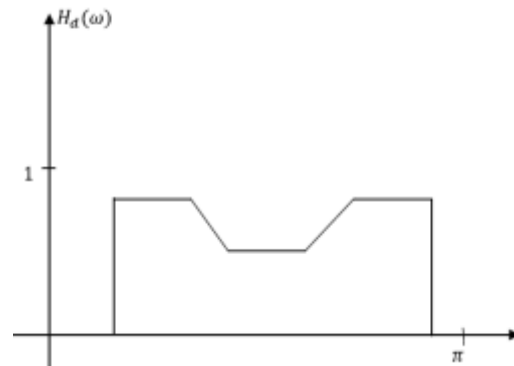As a step 1 we have to Create P number of new sets of weights

Figure 2.   Observation of desired frequency response.

By changing each entry in each parent matrix, with probability p_i, by adding a small random value [4].

As a step 2, now we have to pick a random input target pair from the training set, say (x_i, t_i). Now we have to calculate the output from all children and create the error signal.

As a step 3, We have to recorder the children so that the best children can get the place in top of the position and the rest children are in decreasing order. As a step 4, if the best child

is worse than the parent go to step 1. As a step 5, we should iterate it until no changes in parent [11].

III.   RESEARCH RATIONALE

Genetic Algorithm is optimization tools. As such we can use them in any supervised neural network and also use in unsupervised neural networks, as long as we have error criteria [12]. In this part, we wish to create an FIR filter with an arbitrary transfer function by the use of Genetic Algorithms (GA:s). The FIR filter should be a linear phase filter of odd length. This will give symmetry in the filter taps [3].

We will use a Genetic Algorithm where each parents gives birth to 10 children as a new population. We use the exponential decease function to weight the suitability of becoming a new parent. When initiating the first parent use a pure delay of half the FIR filter length an impulse at position $L - 1/2$ of the FIR filter taps and the rest of the taps at zero [9], [10].

Now we have to create a function where a desired frequency response and the length of the FIR filter are taken as input parameters. The function should then find the least

square or the mini max approximation to the desired transfer function with Genetic Algorithm [7]. When finding the least square or the mini max filter, the only thing that has to be modified is the definition of the error function [8].

We have to remember that when we using the mini max error criteria one should allow for transition regions where the desired frequency response has abrupt changes. Omitting some points around these regions when sampling the frequency does this. Otherwise the maximum error will always appear at these regions. Now we have to use the FIR filter lengths: 33, 65, and 129. The step size and the probability of change, p_i, should be decreased during learning. The constant in the exponentially weighing can be set to 0.1. Sample the frequency region at 1024 points in order to calculate the error function. Now we have to use an appropriate stopping criterion.

IV. DESIGN METHOD OF FIR FILTER

Consider a linear phase FIR filter to approximate any given transfer function. From the (fig. 2) we want to find a L tap FIR filter which have the following transfer function. L- Odd integer number.

We denote the filter taps as $h(n), n = 0, 1, ..., L - 1$. The Fourier transfer of this filter is,

$$H(\omega) = \sum_{n=0}^{L-1} h(n) e^{-j\omega n}$$

(1)

$$H(\omega) = e^{-j\omega \frac{L-1}{2}} \left(h\left(\frac{L-1}{2}\right) + \sum_{n=\frac{L-1}{2}+1}^{L-1} 2h(n) \cos\left(\omega\left(n - \frac{L-1}{2}\right)\right) \right)$$

(2)

It has linear phase and it is symmetric. The absolute value of $H(\omega)$ is,

$$|H(\omega)| = h\left(\frac{L-1}{2}\right) + 2 \sum_{n=\frac{L-1}{2}+1}^{L-1} h(n) \cos(\omega(n - \frac{L-1}{2}))$$

(3)

(since $\left|e^{jx}\right| = 1$). We sample the frequency in $[0, \pi]$ with N points,

$$H_d(\omega) = [H_d(\omega_1), H_d(\omega_2), ..., H_d(\omega_N)]^T$$

(4)

And

$$H(\omega) = [H(\omega_1), H(\omega_2), ..., H(\omega_N)]^T$$

(5)

We can now define an error function,

$$E(H) = \|H_d(\omega) - H(\omega)\|_2^2$$

(6)

TABLE I.   THE NUMBER OF THE FILTER ORDER IMPLEMENTED IN THE NETWORK.

Experiment No.	No. of Filter Order for LS appro.	Implemented Corresponding Functions		Another Observation	
				Filter order for Mini max	Filter order for Band pass filter
1	33	Error Function	Frequency response	33	33
2	65	Error Function	Frequency response	65	65
3	129	Error Function	Frequency response	129	129

Where,

$$h = \left[h\left(\frac{L-1}{2}\right), h\left(\frac{L-1}{2}+1\right), ..., h(L-1) \right]^T$$

(7)

or

$$E(h) = \max|H_d(\omega) - H(\omega)|$$

(8)

We now use the Genetic Algorithm to find the impulse response,

$$h = \left[h\left(\frac{L-1}{2}\right), h\left(\frac{L-1}{2}+1\right), ..., h(L-1) \right]^T$$

(9)

Initialize h to small random values.

V. ANALYSIS OF FINDINGS

According to the table 1 we implemented the number of filter order at LS approximation, Mini max approximation find out the corresponding error function and frequency response. As we mentioned above, we use in here the three filter order 33, 65 and 129 and observe the corresponding function. First we find out the LS approximation with this filter orders and find out the corresponding error function and frequency response function. After that we find out the Mini max approximation with this filter orders.

To choose the faster and efficient approximation between the LS and mini max, we implemented all the filter orders to LS approximation for a band pass filter and we can analysis all the output at the figure 3.

Basically the LS approximation is faster than Mini Max Approximation. But the Step size is also a Factor which can effect on the Convergence.

VI. CONCLUSION

For to do Filter Design and coefficient optimization, we always use the genetic algorithm as efficient and powerful tool. When genetic algorithm designing new filters, it directly optimizes the coefficients by considering the quantization effect. On the other hand, when the genetic algorithm mapping predesigned filter coefficients to integer

arithmetic, since hand quantization does not results in optimal results, it can quickly find a satisfactory result. This genetic algorithm can modify and reshaped as the specific needs and characteristics of the applications.

REFERENCES

[1] Sabbir U. Ahmad and Andreas Antoniou , "Cascade-Form Multiplierless FIR Filter Design Using Orthogonal Genetic Algorithm", IEEE International Symposium on Signal Processing and Information Technology, 2006, pp. 932-937, Aug. 2006

[2] Khadijeh Khamei, Abdolreza Nabavi, Shaahin Hessabi ,"Design Of Variable Fractional Delay Fir Filteirs Using Genetic Algorithm", Proceedings of the 2003 10th IEEE International Conference on Electronics, Circuits and Systems, 2003. ICECS 2003, vol. 1, pp. 48-51, Dec. 2003.

[3] Mehmet Oner and Murat Agkar. "Incremental Design Of High Complexity Fir Filters By Genetic Algorithms", vol.2, pp. 1005-1008, 1999.

[4] Mehmet ONER, "A Genetic Algorithm for Optimisation of Linear Phase FIR Filter Coefficients", Conference on Signals, Systems & Computers, 1998, vol.2, pp. 1397-1400, Nov. 1998.

[5] Paolo Gentili, Fruncesco Piazza and Aurelio Uncini, "Efficient Genetic Algorithm Design For Power-Of-Two Fir Filters", International Conference of Acoustics, Speech, and Signal Processing, 1995. ICASSP-95., vol.2, pp. 1268-1271, May 1995.

[6] D. Suckley ,"Genetic algorithm in the design of FIR filters", IEE Proceedings Circuits, Devices and Systems, vol. 138, pp. 234-238, Apr. 1991

[7] .B. Deng, "Discretization-free design of variable fractional delay FIR filters," IEEE Trans. Circuits Syst. 11: Analog and Digital Signal Processing, June 2001, Vol. 48, No. 6, pp. 637- 644.

[8] C. K. S. Pun, Y. C. Wu, K. L. Ho, "An efficient design offractional delay digital FIR filters using the Farrow structure," Proc. of the 11th EEE Signal Processing Workshop on Statistical Signal Processing, 2001 ,pp. 595 -598.

[9] Y. W. Leung and Y. Wang, "An orthogonal genetic algorithm with quantization for global numerical optimization," IEEE Trans. Evolutionary Comp., vol. 5, no. 1, pp. 41 - 53, Feb. 2001.

[10] J. W. Adams, "FIR digital filters with least-squares stopbands subject to peak-gain constraints," IEEE Trans. Circuits Syst. vol. 39, pp. 376-388, Apr. 1991.

[11] M. Yagyu, A. Nishihara, and N. Fujii, "Fast FIR digital filter structures using minimal number of adders and its application to filter design," IEICE Trans. Fundamentals, vol. E79-A, no. 8, pp. 1120-1128, Aug. 1996.

[12] T.Arslan, D.H.Horrocks, "A genetic algorithm for the design of finite word length arbitrary response cascaded IIR digital filtersLEEE Int. Conf on GA in engineering systems, Sheffield, Sept. 1995, pp 276-281.
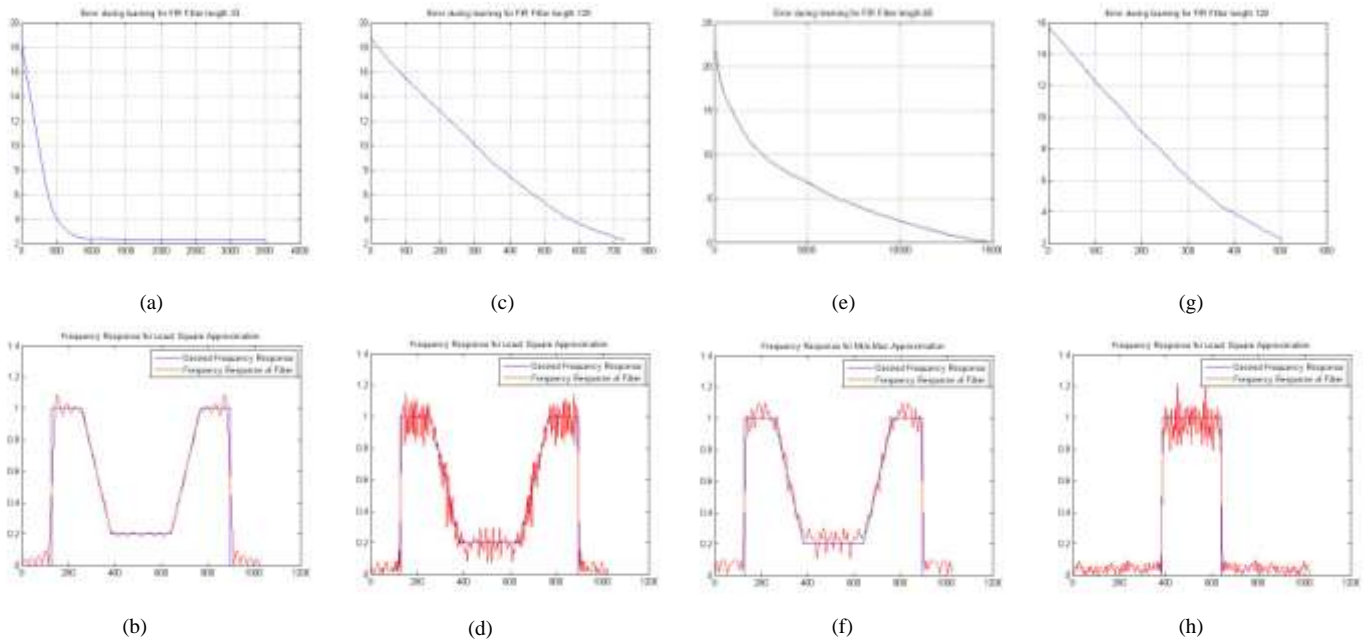
Figure 3. (a) Learning Error for filter order 33, (b) Frequency response for LS approximation for filter order 33, (c) Learning Error for filter order 129, (d) Frequency response for LS approximation for filter order129, (e) Learning Error for filter order 65 in mini max approximation, (f) Frequency response for mini max approximation for filter order 65, (g) Learning Error for filter order 129 for bandpass filter, (h) Frequency response for LS approximation for filter order 129 for bandpass filter.