International Journal of
Numerical Modelling
Electronic Networks, Devices and Fields

Low Power Fractional Delay Sparse Finite Impulse Response Filter Design using Teaching-Learning Based Optimization and its Improved Version

Journal:	*International Journal of Numerical Modelling: Electronic Networks, Devices and Fields*
Manuscript ID	Draft
Wiley - Manuscript type:	Research Article
Date Submitted by the Author:	n/a
Complete List of Authors:	Srivastava, Savita ; Bundelkhand Institute of Engineering and Technology, Electronics and Communication Engg Dwivedi, Atul Kumar; Bundelkhand Institute of Engineering and Technology, Electronics and Communication Engg; Bundelkhand Institute of Engineering and Technology, Jhansi Nagaria, Deepak ; Bundelkhand Institute of Engineering and Technology
Keywords:	FIR Filter, Sparse Filter, Low power Design, Optimization algorithm

Received: Added at production | Revised: Added at production | Accepted: Added at production

DOI: xxx/xxxx

RESEARCH ARTICLE

Low Power Fractional Delay Sparse Finite Impulse Response Filter Design using Teaching-Learning Based Optimization and its Improved Version

Savita Srivastava[1] | **Atul Kumar Dwivedi***[2] | **Deepak Nagaria**[3]

[1]Bundelkhand Institute of Engineering and Technology, Jhansi, U.P, India
[2]Bundelkhand Institute of Engineering and Technology, Jhansi, U.P, India
[3]Bundelkhand Institute of Engineering and Technology, Jhansi, U.P, India

Correspondence
*Atul Kumar Dwivedi. Email: atuldwivedi@live.in

Present Address
Bundelkhand Institute of Engineering and Technology, Jhansi, India

Summary

In signal processing, fractional delay finite impulse response filter is used applications where accuracy, time synchronization, low implementation complexity and stability is prime concern. The aim of sparse finite impulse response filter is to have maximum number of zero value coefficients by minimizing number of non zero value coefficients, which in turn reduces the implementation complexity and the power consumption. However, fractional delay sparse finite impulse response filter design is challenging because of its multimodal,nonuniform and nonlinear objective functions. So, meta heuristic algorithm called teacher learning based optimization algorithm and its improved version has been used here. In this paper, the power consumption in fractional delay sparse finite impulse response filter has been reduced by increasing sparsity and reducing switching activity. The proposed approach is evaluated in three folds. Initially, desired response and actual response of fractional delay sparse finite impulse response filter has been compared. In the second part, proposed method has been compared with other reported methods. The resulting analysis proved that the teaching learning based optimization algorithm shows significant improvement in fractional delay filter design over the other reported methods and also shows minimum deviation from the ideal fractional delay filter. Third part consists of Experimental validation of designed fractional delay sparse finite impulse response filter by implementation on Vertex 7 FPGA by using Xilinx ISE 14.7. The practical aptness is validated in terms of resource utilization and power consumption.

KEYWORDS:
FIR filter; Sparse filter; Sparse Designs; Low power Design; Optimization algorithm; Teaching learning based optimization;

1 | INTRODUCTION

Digital filters are used to have fractional delay in the signal processing applications such as comb filter design [1], speech coding [2], musical instruments modeling [3], software defined radio [4]etc. Fractional delay filters improve accuracy, preciseness, time synchronization of a system. FIR filter can be used for fractional delay applications to improve their stability, simplicity and easiness in implementation [5].

The major challenge of the fractional delay filter is to delay the input signal exactly by a pre specified fractional amount of sampling time [6] without changing any other characteristic.

In literature, various approximation methods are used for the fractional delay filter design [7,8,9] such as weighted least square method [9], minimal and maximal method. Liu et.al. implements the fractional sample delay by using Lagrange interpolation and non-linear interpolation technique [7].

[0]**Abbreviations:** FIR, Finite Inpulse Response; FD, Fractional delay; TLBO, Teaching learning based optimization; DSP, Digital signal processing; IOBS, Input-output Buffer; Bonded IOBS, Input-output Block; karush kuhn tucker, KKT; Abs., Absolute; mini., minimum; maxi., maximum; UB and LB, upper bound and lower bound

Farrow method presents the design of the adjustable delay FIR filter which gives ability to interpolate between the samples [10]. Kumar et.al. presents the variable fractional delay FIR filter design using least square method [9]. Liu et.al. proved that the adjustable weighted least square method is suitable to design variable fractional delay FIR filter than the fixed weighed least square method [11]. Tahar et. al presents the efficient design of Fractional delay FIR filter by using Maclaurin series expansion formula [12]. Lee et. al. designed variable delay filters using hybrid structures and consider the parallel and cascade connection of differentiator simultaneously [13]. However, the major disadvantages of these techniques are that the solution obtained from these methods is locally optimal solution. Because of this, intensive research is required to find the simplest technique to approximate the fractional delay filter.

In recent times, various meta heuristic evolutionary algorithms have been developed which give robust and global optimal solutions which lead to solve optimization problem [14]. These meta heuristic algorithms are capable of solving the complex looking computational problem and solve the large complex problems in lesser time. Some of the heuristic algorithms which are used in the fractional delay filter design are as follows: Genetic algorithm (GA) [14], Particle Swarm Optimisation(PSO) [15], Cuckoo search algorithm (CSA) [16]. Genetic algorithm (GA) which is inspired by Darwin's survival of the fittest technique [17]. Khadijeh et.al. designed variable fractional delay FIR filter using genetic algorithm which results in high speed and low power in his filter implementation [18] . Particle swarm optimisation (PSO) technique is inspired from the flocking behavior of birds. Dongyan et.al present a design of recursive variable fractional delay FIR filter using PSO algorithm [19]. Cuckoo search algorithm (CSA) is based on breeding behavior of the cuckoo bird. Kumar et.al. proved that fractional delay FIR filter design using CSA algorithm is superior than GA, PSO [16].

All evolutionary algorithms based on swarm intelligence are of probabilistic nature, which need common controlling parameters such as number of generations, elite size, population size,etc. These evolutionary algorithms also have their own specific controlling parameters such as crossover rate, mutation rate, cognitive parameters, etc which play cruical role in the performance of algorithm. However, improper tuning of these parameter have limitations like the stagnation problem and he premature convergence. Their implementation results in prohibitive computation complexity. So, to overcome above mentioned problem Teaching Learning Based Optimization algorithm (TLBO) is used for the design of fractional delay FIR filter.

Teaching Learning based optimization (TLBO) algorithm is inspired from teaching learning process [20]. TLBO simulates the standard teaching learning phenomena of the classroom. It is an influential method where each learner aims to learn something from other learner in class. TLBO is based on two phases: teacher phase and learner phase. Teacher phase is learning from the teacher and learning phase is learning from the interaction among the learner [21]. The major advantage of this algorithm is that it does not require any tuning parameter which makes it able to generate better results in a faster convergence rate and simple in implementation unlike the other exiting algorithms. However, in the meta-heuristic evolutionary based computational research there always been an attempt to improve the given algorithm again and again in order to get better results of exploration and exploitation of search space. Thus, TLBO is improved by inclusion of interia weight parameter which helps in solving the problem of global optimization. In addition to that, improved TLBO, i.e.,ITLBO algorithm is analytically proved by Krush Kuhn Tucker optimality conditions[23].

Though a number of designs have been proposed, but none of them have been reported for low power design of sparse FIR filters. Sparse filter design increases the number of zero coefficients. In another way, it minimizes the number of non-zero filter coefficients. In this work, low power design has been achieved by reducing switching activity in FIR filter design. Sparse FIR filter omits the adder and multiplier corresponding to zero valued coefficients [22] which in turn reduces power consumption and implementation, hardware complexity which in turn reduces hardware complexity [23] or by switching activity between filter coefficients [24]. High power consumption at the time of filter execution results in heating of the device which decreases the device life and increases its cooling cost. Also, power minimization is not considered in any existing design of fractional delay sparse FIR filter design using evolutionary algorithms.

In this paper, low power fractional delay sparse FIR filter is designed by using proposed improved teacher learning based optimization(ITLBO). The results achieved by ITLBO has been compared with the TLBO and other reported algorithm and shows significant improvement over the TLBO and other reported algorithm in terms of magnitude error. Further experimental validation for low power consumption and resource utilization is performed by Xilinx system generator and compared in terms of number of slice register, number of LUTS, time delay, switching activity between the filter coefficients. The result obtain after extensive research shows that the proposed method is significantly better than other already exists reported work.

The rest part of the paper is organized as : In section 2, Problem formulation of low power fractional delay FIR filter design is explained. Section 3 described the TLBO algorithm. Section 4 discussed the improved TLBO algorithm and the convergence of improved TLBO algorithm. Section 5 discusses the result analysis of the fractional delay sparse FIR filter using proposed method. Lastly, the conclusion is made for the proposed method.

2 | PROBLEM FORMULATION OF LOW POWER FRACTIONAL DELAY FILTER

The dynamic power consumption in fractional delay sparse FIR filter is directly proportional switching activity$'S'$ $S_{0\leftrightarrow 1}$.

$$P = S_{0\leftrightarrow 1} C V^2 F \tag{1}$$

where F is Frequency , C is Capacitance and V is Voltage swing. Switching activity consists of switching inside the data buses (internal transitions $'S'_{wint}$) and between coupled data buses (coupled transitions $'S'_{wc}$). Therefore the expected value is

$$S_{0\leftrightarrow 1} = \frac{1}{2}(S_{wint} + S_{wc}) \tag{2}$$

The main objective here is to have the optimal filter coefficients to minimize the power consumption and to achieve a desired fractional delay response. A novel objective function is formulated by considering the minimization of maximum error in the frequency response $e(\omega, y)$ including switching activity ($S_{0\leftrightarrow 1}$) by using a Lagrange's multiplier $L(\omega, y)$

$$J(\omega) = Min[Max|e(\omega, y)| + L(\omega, y) S_{0\leftrightarrow 1}] \tag{3}$$

where ω is normalized angular frequency varies in range of $[-\pi, \pi]$. The error between the desired and the actual response is given as

$$e(\omega, y) = ||H_{id}(\omega, y)| - |H_a(\omega, y)|| \tag{4}$$

where $H_{id}(\omega, y)$ is frequency response of the ideal FD filter and $H_a(\omega, y)$ is same for the actual designed FD filter.

The frequency response of ideal FD filter [25] is represented as

$$H_{id}(\omega, y) = \exp^{-j\omega y} \tag{5}$$

where y is a fractional delay variable in [-0.5, 0.5] range. The transfer function of FIR filter which is used to approximate the desired specifications is given as

$$H(b, y) = \sum_{j=0}^{N} h(j, y) b^{-j} \tag{6}$$

here $h(j, y)$ is real valued variable, N is filter order. Using Taylor series expansion at y=0, $h(j, y)$ coefficient is given as

$$h(j, y) = \sum_{k=0}^{M} a(j, k) y^k \tag{7}$$

By putting (3) in (2) the transfer function can be rewritten as,

$$H(b, y) = \sum_{j=0}^{N} \sum_{k=0}^{M} a(j, k) y^k b^{-j} = \sum_{k=0}^{M} A(b, k) y^k \tag{8}$$

So, the frequency response of H(z,y) is written as

$$H_a(\omega, y) = \sum_{k=0}^{M} A(k, \omega) y^k \tag{9}$$

To minimize power consumption and magnitude response error the objective problem (3)has been solved using TLBO algorithm and its improved version.

3 | TEACHING LEARNING BASED OPTIMIZATION ALGORITHM

Teaching learning is nature inspired population based algorithm that follow the teaching and learning process [26]. It is an influential method which aims to improve themselves by learning something from other people. In TLBO, the possible solution of the optimization problem is to consider as the learner. The global best solution is taken as a teacher. From the each iteration process, the new teacher is selected. The population size is analogous to the number of learners, the parameters of the design problem are based on the number of subjects learned by the learners [26]. The basic idea of this algorithm is to enhance the knowledge of the learner by improved its parameter which lead to give the optimal solution. This algorithm has two fundamental phases of the learning 1. Teacher phase 2. Learner phase. Teacher phase: In this phase, the learner interacts with teachers to improve their knowledge. In a classroom, the teachers quality influence the knowledge of the student. The best solution of the set is elected as teacher who share their knowledge with learner in order to improve the mean of the classroom [27]. Suppose m is iterations, 'd' is design variables or number of subjects, l is number of learners population size(n=1,2,. . . ,l). The position of each learner in the present iteration is given as

$$q_{d,nNew,m} = q_{d,nOld,m} + randi * (q_d, nbest, m - T_F * q_{mean_{d,m}}) \tag{10}$$

where $q_{d,nNew,m}$ and $q_{d,nOld,m}$ is the new and old positions of the learner at m^{th} iteration, randi is random vector distribution uniformly in the range of [0, 1] ,$q_d, nbest, m$ is best teacher (i.e. learner) ,$q_{mean_{d,m}}$ is the mean value of the teacher, T_F is teaching factor which is expressed as :

$$T_F = round[1 + rand(0-1)\{2-1\}] \tag{11}$$

The value T_F is either 1 or 2, which is randomly selected with equal probability based on the rounding up criteria as given in 11. In TLBO algorithm, T_F is used as adjusting factor for controlling the moving direction during the solution updating process. T_F is not considered as specific TLBO algorithms parameter.

Learner Phase: In learning phase, students share their knowledge with the other fellow students. They randomly grouped with other learners to enhance their grades [27]. In this phase two students q_a and q_b are not equal $(i.e. a \neq b)$. The learner phase for m^{th} parameter is given as:

$$q_{d,aNew,m} = q_{d,aOld,m} + rand(q_{d,a,m} - q_{d,b,m}) \tag{12}$$

else

$$q_{d,aNew,m} = q_{d,aOld,m} + rand(q_{d,b,m} - q_{d,a,m}) \tag{13}$$

where $q_{d,aNew,m}$ and $q_{d,aOld,m}$ are the a^{th} and b^{th} position of the learner respectively, rand is random number in [0 1] range.

Psuedocode 1

Randomly Initialize the Number of population, Max iterations
while (m $< m_{Max}$ Max Generation)
Determine the fitness function for each individual
Update the rank of each individual to determine the teacher (T_m)
for m= 1 to D
Calculate the value of $q_{d,nNew,m}$ by using eq. (14) and eq. (11)
Update the value of $q_{d,nNew,m}$ if f($q_{d,nNew,m}$) is better than f($q_{d,nOld,m}$)
end for
for m= 1 to D
Randomly select two students q_a and q_b i.e. $(a \neq b)$
if $q_b < q_a$
Update the value of $q_{d,aNew,m}$ by using eq.(15)
else
Update the value of $q_{d,aNew,m}$ by using eq.(16)
end
end for
m = m+1
Update the current best solution observed so far
end while

4 | IMPROVED TEACHING LEARNING BASED OPTIMIZATION ALGORITHM

The TLBO method is an iterative process in which continuous interaction taking place for transferring the knowledge [28]. The students are trying to enhance their knowledge by learning new things from the teacher [29]. In the iterative, the exploration and exploitation process must be balanced. Exploration means ability to explore the different search areas in order to find global solutions. Exploitation means ability to focus the search area around a promising space in order to find optimum solution more precisely. So, a parameter called interia weight 'w' is added to the teacher and learner phase. The weight linearly decreases in order to make a balance between the exploration and exploitation process of the learner and teacher to get an optimum global solution. The improved teacher phase and can be written as

$$q_{d,nNew,m} = w * q_{d,nOld,m} + randi * (q_d, nbest, m - T_F * q_{mean_{d,m}}) \tag{14}$$

The improved learner phase can be written as

$$q_{d,aNew,m} = w * q_{d,aOld,m} + rand(q_{d,a,m} - q_{d,b,m}) \tag{15}$$

else

$$q_{d,aNew,m} = w * q_{d,aOld,m} + rand(q_{d,b,m} - q_{d,a,m}) \tag{16}$$

where, ω is interia weight which is given as

$$w = w_{max} - \frac{w_{max} - w_{min}}{itr_{max}} * itr \tag{17}$$

where, ω is the inertia weight at current iteration t, w_{max} is initial value of the inertia weight, w_{min} is final value of the inertia weight.

4.1 | Convergence of Improved Teaching Learning Based Optimization Algorithm

In this subsection, the convergence of improved TLBO algorithm has been proved analytically by using Karush Kuhn Tucker optimal condition [30] .

$$minimize : f(q) \tag{18}$$

$$subject \quad to : (q - u) \leq 0 \;\; and \;\; (-q + 1) \leq 0 \tag{19}$$

where l and u reprsents lower and upper bound limit The KKT optimal condition at point q εf for the first order are expressed as

$$\triangle f(q) + \alpha_1 + \alpha_2 = 0 \tag{20}$$

$$\alpha_1(c - l) \leq 0 \;\; and \;\; \alpha_2(-c + 1) \leq 0 \tag{21}$$

where $\alpha_1 \leq 0$ and $\alpha_2 \leq 0$ $|\alpha_1, \alpha_2 \;\; \varepsilon R^n|$

The other KKT conditions are given as :

1. If $m_i < q_i < l_i$, $\Rightarrow (\bigtriangledown f(q))_i = 0$.
2. If $q_i = l_i$ and $m_i < q_i$, then $(\alpha_2)_i$ = 0, $\Rightarrow (\bigtriangledown f(q))_i$ = $(\alpha_1)_i$ $\equiv (\bigtriangledown f(q))_i \leq 0$
3. If $q_i = m_i, q_i < l_i$, then $(\alpha_2)_i = 0$, $\Rightarrow (\bigtriangledown f(q))_i$ = $(\alpha_2)_i \equiv (\bigtriangledown f(q))_i \leq 0$

The Taylor series expansion of $f(q)$ at $point$ (q_k) is written as

$$f(q) = f(q_k) + (q - q_k) \triangle f(k) + \frac{q - q_k}{2!} H_m (q - q_k)^T + \ldots\ldots \tag{22}$$

where H_m represents hessian matrix.

The first two terms have expanded on

$$\triangle f(q) = \triangle f(q_k) + (q - q_k) \tag{23}$$

$$\triangle f(q) = \triangle f(q_k) + (q - q_k) H_m \tag{24}$$

If $f(q)$ in range $m < (q*) < l$, is optimal solution of $(q*)$, then by KKT condition, $(\triangle f(q) = 0)$

$$\triangle f(k) + (q - q_k) H_m = 0 \tag{25}$$

$$q = f(q_k) - H_m^{-1} \triangle f(q_k) \tag{26}$$

At initial point (q_k), the next point is computed as:

$$q_k + 1 = q_k + \mu_k H_m k \triangle f(q_k) \tag{27}$$

where (q_k) is step size. $(H_m k)$ is hessian approximated matrix at point 'k'.

The fitness function corresponding to it is expressed as

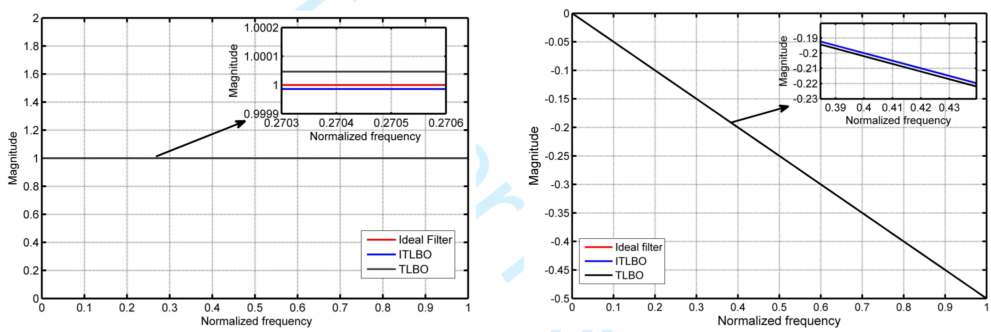
$$f(u_{ij}(x)) = f(q_{ij}(x)) + H_m \times \triangle f \tag{28}$$

where H_m is the hessian approximated matrix which produces the disruption $H \times \triangle f(k)$ in the fitness $f(q_{ij}(x))$.

The every process iteration assured the myriad improvement in convergence of the fitness function of improved TLBO algorithm to the optimum point. The results obtained post convergence have been outlined in the next section.

TABLE 1 Optimized coefficients of FD FIR filter

Filter length	Filter coefficients	TLBO	ITLBO
	h(0)	0.0000000000000000	0.0000000000000000
3	h(1)	0.9992322138241350	0.9997748543858380
	h(2)	0.0000000000000000	0.0000000000000000
	h(0)	0.0000000000000000	0.0000000000000000
	h(1)	0.0000197464980277	0.9999997886525600
4	h(2)	0.0000000000000000	0.0000000000000000
	h(3)	-1.0000023252260700	1.16932419039488E-06
	h(0)	0.0000000000000000	0.0000000000000000
	h(1)	0.9999997344963110	-0.999999997749532
5	h(2)	0.0000000000000000	0.0000000000000000
	h(3)	-2.56429273598981E-07	-6.16790087842628E-09
	h(4)	1.36739574968598E-07	4.95313057534883E-09



(a) Magnitude response **(b)** Phase response

FIGURE 1 Normalized response of FD sparse FIR filter of filter length '3'

5 | RESULT ANALYSIS

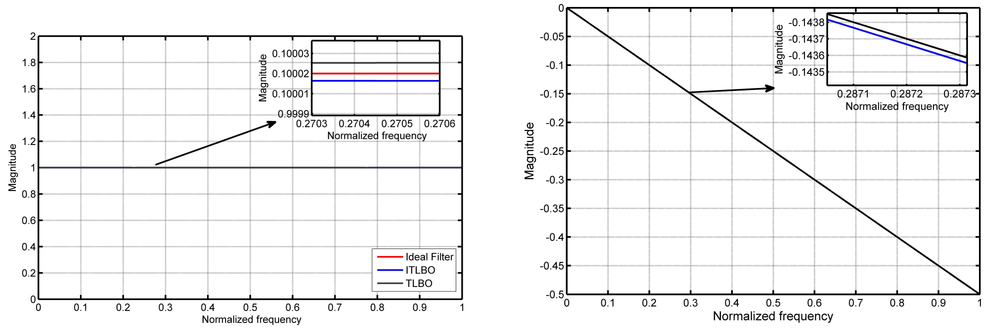
The fruition of the proposed method in the design of low power fractional delay sparse FIR filter has been evaluated by comparing it with other reported methods in the context to error minimization between the desired and the ideal response of the FD sparse FIR filter. The design steps for fractional delay sparse FIR filter has been discussed in the previous sections. The control parameters of the proposed TLBO algorithm are set as: UB and LB= [−10, 10], population size =25 and iterations= 200. The filter length is '3', '4', '5'.

The effectiveness of the proposed low power fractional delay sparse FIR filter design using ITLBO algorithm has been evaluated in three parts. In the first part, magnitude response of the fractional delay sparse FIR filter has been analyzed in comparison to ideal filter. Second part, elaborate the proposed ITLBO algorithm effectiveness in the design of fractional delay sparse FIR filter in terms of absolute magnitude error over the standard TLBO and the other reported algorithms. Third part, consists of the hardware implementation of the designed fractional delay sparse FIR filter on Virtex-7 with Xilinx ISE 14.7.

The filter coefficients obtained after convergence of proposed TLBO and ITLBO algorithms have filter length '3' with 66.66% sparsity , filter length '4' with 50.00% sparsity and filter length '5' with 40.00% sparsity, which has been shown in Table 1 . By using proposed ITLBO, the magnitude response of FD sparse FIR filter has been compared with TLBO and ideal filter. The magnitude response comparison plot of filter length '3', '4' and '5' has been shown in figure (1-3). The absolute magnitude error and absolute phase error comparison plot of filter length '3', '4' and '5' has been shown in figure (4-6). To the clarity of the overlapped filter response, a zoomed plot is also shown in these figures. It can be seen from magnitude plots that the ITLBO is approximated more near to the ideal filter than the TLBO.

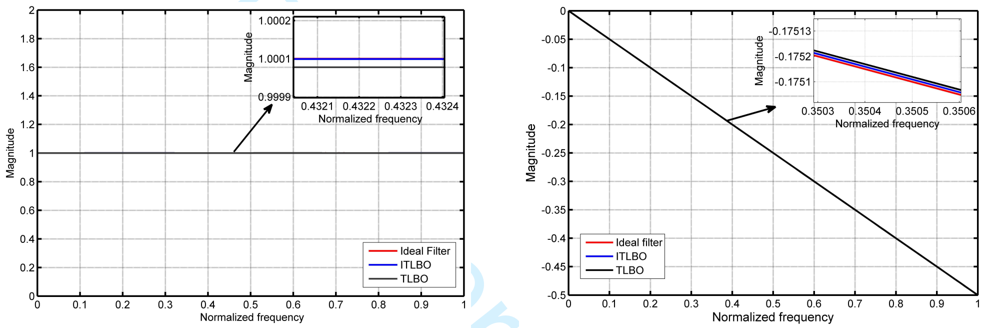
TABLE 2 Magnitude error comparison of FD sparse FIR filter design using ITLBO with ITLBO and other reported methods

Methods	Filter length	Sparsity(%)	Magnitude Error		
			Minimum	Maximum	Average
DHT[31]	80	0.00	3.5420×10^{-3}	-	-
DFT[32]	80	0.00	2.8410×10^{-3}	-	-
DCT-II[33]	80	0.00	2.5240×10^{-3}	-	-
RBF[34]	60	0.00	1.8320×10^{-3}	-	-
GA[16]		0.00	2.7368×10^{-4}	2.7887×10^{-2}	1.0140×10^{-2}
PSO[35]		0.00	1.1240×10^{-4}	2.8430×10^{-3}	1.4770×10^{-3}
CSA[16]	3	0.00	5.1013×10^{-5}	2.6654×10^{-3}	1.3580×10^{-3}
IFFA[35]		0.00	1.5000×10^{-5}	3.8000×10^{-5}	2.6501×10^{-5}
TLBO		66.66	4.6908×10^{-5}	7.6678×10^{-4}	4.0684×10^{-4}
ITLBO		66.66	1.3756×10^{-5}	1.9216×10^{-5}	1.6486×10^{-5}
GA[16]		0.00	9.5026×10^{-5}	6.2456×10^{-3}	1.3293×10^{-3}
PSO[35]		0.00	7.5000×10^{-5}	1.6880×10^{-3}	1.2190×10^{-4}
CSA[16]	4	0.00	4.1839×10^{-5}	1.9008×10^{-3}	6.3157×10^{-4}
IFFA[35]		0.00	6.0240×10^{-6}	2.6109×10^{-5}	1.6014×10^{-5}
TLBO		50.00	2.3263×10^{-6}	5.4942×10^{-6}	3.9102×10^{-6}
ITLBO		50.00	1.3806×10^{-6}	1.5442×10^{-6}	1.4624×10^{-6}
GA[16]		0.00	9.360×10^{-5}	1.206×10^{-3}	1.0840×10^{-3}
PSO[35]		0.00	2.2341×10^{-5}	3.143×10^{-4}	3.124×10^{-4}
CSA[16]	5	0.00	1.3300×10^{-5}	4.1020×10^{-4}	2.1026×10^{-4}
IFFA[35]		0.00	1.1000×10^{-6}	1.34000×10^{-6}	1.2100×10^{-6}
TLBO		40.00	6.5863×10^{-7}	8.6184×10^{-7}	7.6023×10^{-7}
ITLBO		40.00	1.1266×10^{-8}	1.7160×10^{-8}	1.4213×10^{-8}



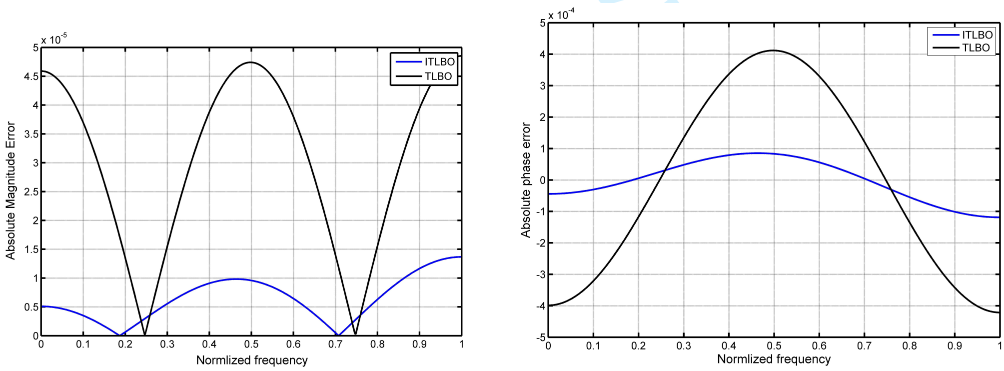
(a) Magnitude response **(b)** Phase response

FIGURE 2 Normalized response of FD sparse FIR filter of filter length '4'



(a) Magnitude response **(b)** Phase response

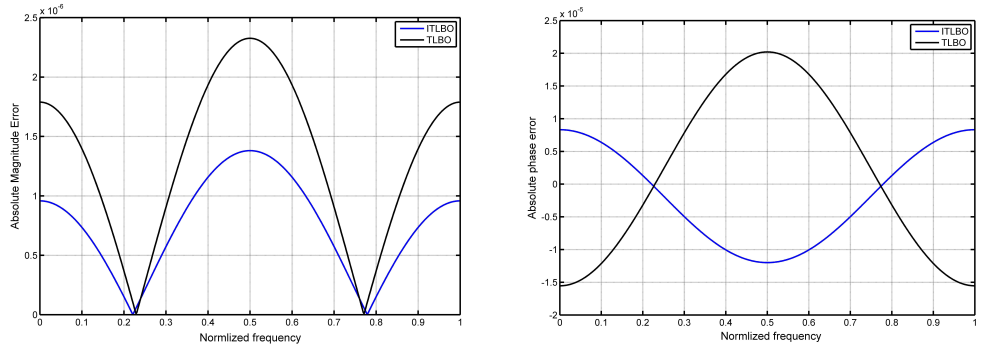
FIGURE 3 Normalized response of FD sparse FIR filter of filter length '5'



(a) Abs. magnitude error **(b)** Abs. phase error

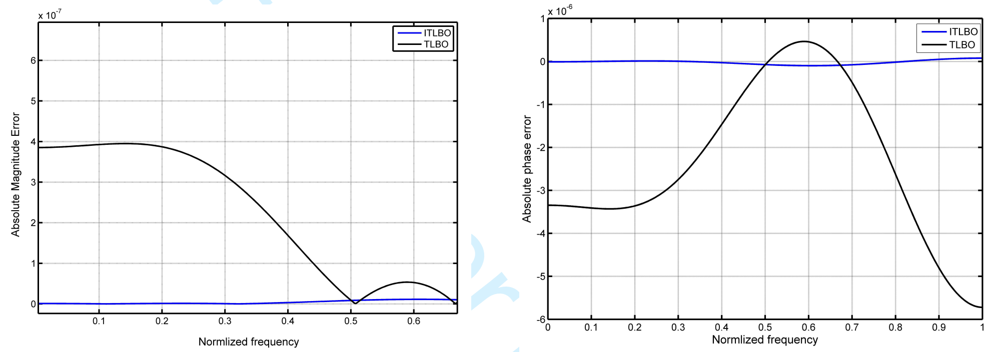
FIGURE 4 Abs. magnitude error and Abs. phase error of FD FIR filter of filter length '3'

The proposed TLBO and ITLBO algorithm compared with existing algorithms, i.e., DHT [31], DFT [32], DCT-II [33], RBF [34], GA [16],PSO [35], CSA [16], has been shown in Table 2 in terms of maxi. magnitude error, mini. magnitude error and average of magnitude error for filter length '80', '60', '3', '4' and '5'. Tseng et.al. used DHT method to design a FD FIR filter of filter length 80 and shown the magnitude error for 3.5420×10^{-3}, Tseng et.al. applied DFT method of filter length '80' and found the magnitude error of 2.8410×10^{-3} , Tseng et.al. implemented DCT-11 method for filter length '80' and obtained the magnitude error of 2.8410×10^{-3} , Tseng et.al. achieved the magnitude error of 1.8320×10^{-3} filter length '60' by using the RBF method, Kumar et.al. achieved the max. error of 2.7887×10^{-2} and mini. error 2.7368×10^{-4} for filter length '3', maxi. error of 6.2456×10^{-3}



(a) Abs. magnitude error **(b)** Abs. phase error

FIGURE 5 Abs. magnitude error and Abs. phase error of FD FIR filter of filter length '4'



(a) Abs. magnitude error **(b)** Abs. phase error

FIGURE 6 Abs. magnitude error and Abs. phase error of FD FIR filter of filter length '5'

and mini. error 9.5026×10^{-5} for filter length '4', maxi. error 1.206×10^{-3} and mini. error 9.360×10^{-5} for filter length '5' by using the GA method. Srivastava.S et.al. achieved the maxi. error of 2.8430×10^{-3} and mini. error 1.1240×10^{-4} for filter length '3', maxi. error of 1.6880×10^{-3} and mini. error 7.5000×10^{-5} for filter length '4', maxi. error of 3.4130×10^{-4} and mini. error 2.2341×10^{-5} for filter length '5' by using the PSO method. Kumar et.al. achieved the maxi. error of 2.6654×10^{-3} and mini. error 5.1013×10^{-5} for filter length '3', maxi. error 1.9008×10^{-3} and mini. error 4.1839×10^{-5} for filter length '4', maxi. error 4.1020×10^{-4} and mini. error 1.3300×10^{-5} for filter length '5' by using the CSA method.

For TLBO with filter length '3', the maxi. magnitude error is 7.6678×10^{-4} and mini. magnitude error is 4.6908×10^{-5} and for ITLBO, the maxi. magnitude error is 1.9216×10^{-5} and mini. magnitude error is 1.3756×10^{-5} . In TLBO with filter length '4', the maxi. magnitude error is 5.4942×10^{-6} and mini. magnitude error is 2.3263×10^{-6} and for ITLBO, maxi. magnitude error is 1.5442×10^{-6} and mini. magnitude error is 1.3806×10^{-6} . For TLBO with filter length '5', maxi. magnitude error is 8.6184×10^{-7} and mini. magnitude error is 6.5863×10^{-7} and for ITLBO, maxi. magnitude error is 1.7160×10^{-8} and mini. magnitude error is 1.1266×10^{-8}.

From Table 2 it is noticeable that the FD sparse FIR filter design with proposed method has better performance in terms of magniude error. Therefore, the proposed method performs better than the TLBO and Other reported methods in the context of magnitude error. Table 3 represents the percentage improvement using proposed ITLBO algorithm in comparison to DHT [31], DFT [32], DCT-II [33], RBF [34], GA [16],PSO [35], CSA [16] and TLBO algorithm. It can be seen from the table that proporsed method over TLBO shows 70.674% of improvement for filter length '3', 40.652% improvement for filter length '4' and 98.289% of improvement improvement for filter length '5' in minimum error. For maximum error, proporsed method shows the percentage improvement of 97.493% for filter length '3', 71.715% for filter length '4', 98.089% for filter length '5' over TLBO. It is noticeable from the table 3 that the proposed method shows highest maximum percentage improvement in magnitude error over DHT, DFT, DCT-II,RBF, GA, PSO, CSA and TLBO algorithms. As the filter length increases from '3' to '5', percentage improvement increases in maximum magnitude error. After analyzing Table 2 and Table 3, it is concluded that the proposed algorithm shows significant improvement over the TLBO and other reported methods in terms of magnitude error.

TABLE 3 Percentage improvement in magnitude error of FD sparse FIR filter using proposed ITLBO algorithm with respect to other reported method

Methods	Filter length	Percentage Improvement		
		Minimum	Maximum	Average
DHT [31]	80	99.611	-	-
DFT [32]	80	99.515	-	-
DCT-II [33]	80	99.454	-	-
RBF [34]	60	99.249	-	-
GA [16]	3	94.973	99.931	98.713
PSO		87.761	99.324	93.542
CSA [16]		73.027	99.279	98.406
IFFA [16]		8.293	49.431	28.862
TLBO		70.674	97.493	95.958
GA [16]	4	98.547	99.975	98.796
PSO [35]		98.160	99.908	99.034
CSA [16]		96.700	99.918	97.466
IFFA [35]		77.091	94.088	85.589
TLBO		40.652	71.715	62.473
GA [16]	5	99.987	99.998	99.909
PSO [16]		99.949	99.990	99.969
CSA [16]		99.915	99.995	99.915
IFFA [16]		98.981	98.723	98.852
TLBO		98.289	98.089	80.000

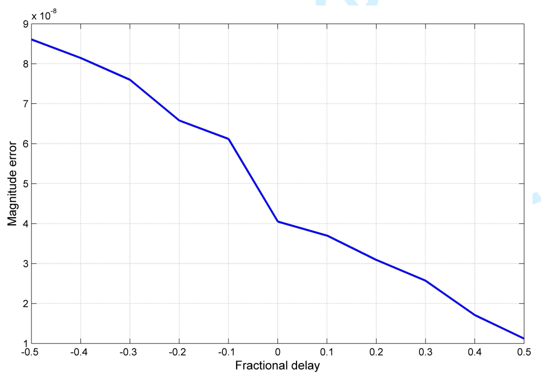


FIGURE 7 Magnitude error at different FD sparse FIR filter variable using proposed ITLBO algorithm with filter length '5'

Table 4 shows the performance of the fractional delay sparse filter of filter length 5 by changing the fractional delay variable in range [-0.5, 0.5]. It is observed that the fractional delay variables linearly decreases from -0.5 to 0.5. The variation in magnitude error with the fractional delay [-0.5, 0.5] is depicted in figure 7 and figure 8.

Furthermore, Table 5 shows the effect of sparsity on fractional delay sparse filter. It can be observed that with increasing the sparsity or having maximum number of zero coefficients, magnitude error decreases considerably and the location of the coefficients has no effect on magnitude error.

TABLE 4 Magnitude error at different FD variables using proposed ITLBO algorithm

Fractional delay	Filter length	Magnitude Error		
		Minimum	Maximum	Average
	3	8.8156×10^{-5}	9.2830×10^{-5}	9.0493×10^{-5}
-0.5	4	9.4619×10^{-6}	9.6585×10^{-6}	9.5602×10^{-6}
	5	8.6184×10^{-8}	8.8032×10^{-8}	8.7108×10^{-8}
	3	8.2758×10^{-5}	8.5794×10^{-5}	8.4276×10^{-5}
-0.4	4	8.2886×10^{-6}	8.6768×10^{-6}	8.4827×10^{-6}
	5	8.1453×10^{-8}	8.2587×10^{-8}	8.2020×10^{-8}
	3	7.2100×10^{-5}	7.3152×10^{-5}	7.2626×10^{-5}
-0.3	4	7.5379×10^{-6}	7.6803×10^{-6}	7.6091×10^{-6}
	5	7.6020×10^{-8}	7.7975×10^{-8}	7.6997×10^{-8}
	3	7.4403×10^{-5}	7.6750×10^{-5}	7.5576×10^{-5}
-0.2	4	6.6496×10^{-6}	6.8668×10^{-6}	6.7391×10^{-6}
	5	6.5828×10^{-8}	6.8955×10^{-8}	6.7391×10^{-8}
	3	6.5828×10^{-5}	6.8955×10^{-5}	6.7391×10^{-5}
-0.1	4	6.2830×10^{-6}	6.5318×10^{-6}	6.4074×10^{-6}
	5	6.1210×10^{-8}	6.2580×10^{-8}	6.1895×10^{-8}
	3	5.2538×10^{-5}	5.5582×10^{-5}	5.4060×10^{-5}
0.0	4	5.0542×10^{-6}	5.4627×10^{-6}	5.2584×10^{-6}
	5	4.0508×10^{-8}	4.6212×10^{-8}	4.3360×10^{-8}
	3	4.1268×10^{-5}	4.3221×10^{-5}	3.7366×10^{-8}
0.1	4	4.6318×10^{-6}	4.7761×10^{-6}	3.7366×10^{-6}
	5	3.7095×10^{-8}	3.7638×10^{-8}	3.7366×10^{-8}
	3	3.1731×10^{-5}	3.6207×10^{-5}	3.3969×10^{-5}
0.2	4	3.2398×10^{-6}	3.3976×10^{-6}	3.3187×10^{-6}
	5	3.0966×10^{-8}	3.1490×10^{-8}	3.1228×10^{-8}
	3	2.6214×10^{-5}	2.7026×10^{-5}	2.5973×10^{-5}
0.3	4	2.2228×10^{-6}	2.4940×10^{-6}	2.3584×10^{-6}
	5	2.5742×10^{-8}	2.6204×10^{-8}	2.5973×10^{-8}
	3	2.1523×10^{-5}	2.2251×10^{-5}	2.2018×10^{-5}
0.4	4	1.9768×10^{-6}	2.096×10^{-6}	2.0364×10^{-6}
	5	1.9576×10^{-8}	2.0160×10^{-8}	1.8368×10^{-8}
	3	1.3756×10^{-5}	1.9216×10^{-5}	1.6486×10^{-5}
0.5	4	1.3806×10^{-6}	1.5442×10^{-6}	1.2220×10^{-8}
	5	1.1268×10^{-8}	1.7160×10^{-8}	1.2220×10^{-8}

The practical aptness designed filter using proposed method has been evaluated by implementing it on hardware Virtex-7 FPGA using Xilinx ISE 14.7. Figure 9 depicts hardware experimental setup of proposed fractional delay sparse FIR filter design implementation of filter length '5' using Virtex-7.

Table 6 presents a hardware resources utilization report of FD sparse FIR filter over FD non sparse FIR filter [35] in the context to number of- slice register, bonded IOBS, IOBS slice LUTS, and switching activity for filter length '5'.

It shows the significant improvement in fractional delay sparse FIR filter over traditional fractional delay non sparse FIR filter design. Hence, the proposed fractional delay sparse FIR filter design can be used in applications where high speed, less implementation complexity, less implementation cost and low power consumption required.

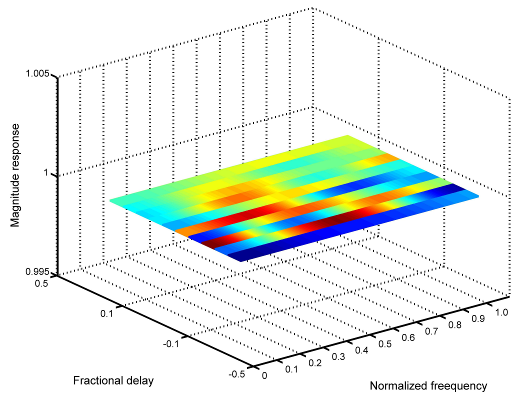


FIGURE 8 Three dimensional view of Magnitude error at different FD sparse FIR filter variable using proposed ITLBO algorithm in filter length 5

TABLE 5 Sparsity report of magnitude error of FD sparse FIR filter using proposed ITLBO algorithm

Filter length	Sparsity	Minimum error	Maximum error	Average
3	33.33	5.0562×10^{-5}	1.1170×10^{-3}	6.1028×10^{-4}
	66.66	1.3756×10^{-5}	1.9216×10^{-5}	1.6486×10^{-5}
4	25.00	1.3555×10^{-5}	2.2918×10^{-4}	1.3236×10^{-4}
	50.00	1.3806×10^{-6}	1.5442×10^{-6}	1.4624×10^{-6}
5	20.00	6.6496×10^{-6}	9.8881×10^{-6}	8.2688×10^{-6}
	40.00	1.1266×10^{-8}	1.7160×10^{-8}	1.4213×10^{-8}
	60.00	1.9716×10^{-9}	2.7469×10^{-9}	2.3614×10^{-9}

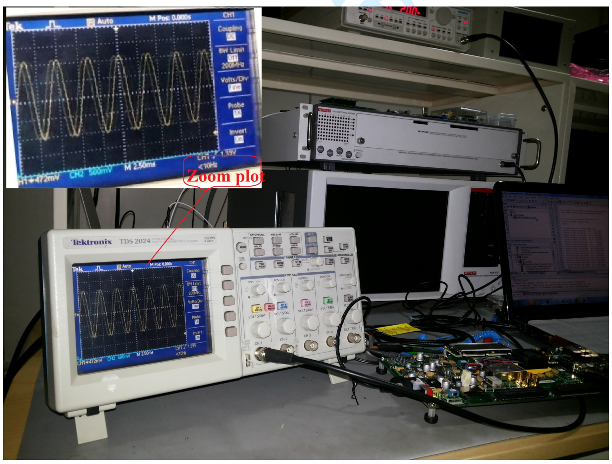


FIGURE 9 Hardware experimental set up of FD sparse FIR filter implementation using proposed ITLBO algorithm

6 | CONCLUSION

In this paper, a novel meta heuristic evolutionary technique ITLBO algorithm with its traditional version, is used to optimized the cost function which is used to design low power fractional delay FIR filter. Considering the low power objective, the design was sparse, which has minimum number of nonzero coefficients that reduces the implementation cost and complexity of the design. The proposed ITLBO algorithm was compared with already existing evolutionary algorithm GA and CSA and the other reported conventional method DHT, DFT, DCT-11, RBF, PSO. For real time signal processing applications, efficiency of the proposed designed method was also tested in terms of resource utilization and low power

TABLE 6 Device utilization report of FD sparse FIR filter coefficients using proposed ITLBO algorithm for filter length 5

Parameters	No. of the devices utilized	
	Sparse	Reported method [35]
Slice register	37	-
Slice LUTS	32	45
Bonded IOBS	46	48
Time delay(nsec)	1.06	1.26
IOBS	45	47
Switching activity	29	-

consumption. The obtained results depicted that the proposed ITLBO algorithm shows sufficient improvement than the standard TLBO and other conventional methods. The main reason which makes ITLBO more effective is having less controlling parameter for tuning the algorithm. Thus, the proposed ITLBO algorithm is less complex and simpler in implementation. The proposed designs have also been validated using hardware implementation over the Virtex-7 FPGA.

7 | ACKNOWLEDGEMENT

This work is supported by the Homi Bhabha research scheme of Dr. A. P. J. Abdul Kalam Technical University, Lucknow. Further, the authors would like to thank Prof R.M. Patrikar, Professor VNIT Nagpur, India for providing the necessary hardware implementation support.

References

1. Pei SC, Huang YD, Lin SH, Shyu JJ. Design of variable comb filter using FIR variable fractional delay element. *Signal Processing* 2012; 92(10): 2409–2421.

2. Marques J, Trancoso I, Tribolet J, Almeida L. Improved pitch prediction with fractional delays in CELP coding. In: IEEE. ; 1990: 665–668.

3. Pakarinen J, Karjalainen M, Valimaki V, Bilbao S. Energy behavior in time-varying fractional delay filters for physical modeling synthesis of musical instruments. In: . 3. IEEE. ; 2005: iii–1.

4. Li H, Van Kerrebrouck J, Bauwelinck J, Demeester P, Torfs G. Maximally Flat and Least-Square Co-Design of Variable Fractional Delay Filters for Wideband Software-Defined Radio. *Journal of Circuits, Systems and Computers* 2019; 28(01): 1950006.

5. Dwivedi AK, Ghosh S, Londhe ND. Modified artificial bee colony optimisation based FIR filter design with experimental validation using field-programmable gate array. *IET Signal Processing* 2016; 10(8): 955–964.

6. Nguyen HT, Do MN. Hybrid filter banks with fractional delays: Minimax design and application to multichannel sampling. *IEEE transactions on signal processing* 2008; 56(7): 3180–3190.

7. Liu G, Wei C. Programmable fractional sample delay filter with Lagrange interpolation. *Electronics Letters* 1990; 26(19): 1608–1610.

8. Deng TB, Lu WS. Weighted least-squares method for designing variable fractional delay 2-D FIR digital filters. *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing* 2000; 47(2): 114–124.

9. Kumar M, Rawat TK. Design of a variable fractional delay filter using comprehensive least square method encompassing all delay values. *Journal of Circuits, Systems and Computers* 2015; 24(08): 1550116.

10. Eghbali A, Johansson H, Saramäki T. A method for the design of Farrow-structure based variable fractional-delay FIR filters. *Signal Processing* 2013; 93(5): 1341–1348.

11. Lu WS, Deng TB. An improved weighted least-squares design for variable fractional delay FIR filters. *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing* 1999; 46(8): 1035–1040.

12. Bensouici T, Charef A, Assadi I. A new approach for the design of fractional delay by an FIR filter. *ISA transactions* 2018; 82: 73–78.

13. Lee SL. Hybrid parallel/cascade structure for designing variable fractional-delay filters. *IET signal processing* 2012; 6(7): 661–672.

14. Ababneh JI, Bataineh MH. Linear phase FIR filter design using particle swarm optimization and genetic algorithms. *Digital Signal Processing* 2008; 18(4): 657–668.

15. Bansal JC, Singh P, Saraswat M, Verma A, Jadon SS, Abraham A. Inertia weight strategies in particle swarm optimization. In: IEEE. ; 2011: 633–640.

16. Kumar M, Rawat TK. Optimal design of fractional Delay FIR filter using cuckoo search algorithm. *International Journal of Circuit Theory and Applications* 2018; 46(12): 1–16.

17. Ahmad SU, Antoniou A. A genetic algorithm approach for fractional delay FIR filters. In: IEEE. ; 2006: 4.

18. Khamei K, Nabavi A, Hessabi S, Barandagh SAM. Design of variable fractional delay FIR filters with CSD coefficients using genetic algorithm. *Journal of Circuits, Systems, and Computers* 2005; 14(06): 1145–1155.

19. Sun D, Zhao J, Zhao X. A recursive variable fractional delay FIR filters design scheme using particle swarm optimization algorithm. In: . 1. IEEE. ; 2009: 109–113.

20. Mohanty B, Tripathy S. A teaching learning based optimization technique for optimal location and size of DG in distribution network. *journal of electrical systems and information technology* 2016; 3(1): 33–44.

21. Chen X, Xu B, Yu K, Du W. Teaching-Learning-Based Optimization with Learning Enthusiasm Mechanism and Its Application in Chemical Engineering. *Journal of Applied Mathematics* 2018; 2018.

22. Ye W, Yu YJ. Greedy algorithm for the design of linear-phase FIR filters with sparse coefficients. *Circuits, Systems, and Signal Processing* 2016; 35(4): 1427–1436.

23. Aktan M, Yurdakul A, Dundar G. An algorithm for the design of low-power hardware-efficient FIR filters. *IEEE Transactions on Circuits and Systems I: Regular Papers* 2008; 55(6): 1536–1545.

24. Rahmeier JG, Luz ÂG, Costa EC, Ghissoni S. Reducing switching activity in FIR filters by reordering the coefficients through the use of improved heuristic algorithm. In: IEEE. ; 2013: 33–36.

25. Deng TB. Weighted-least-squares design of odd-order variable fractional-delay filters using coefficient-symmetry. In: IEEE. ; 2007: 1–5.

26. Qu D, Liu S, Zhang D, Wang J, Gao C. Teaching-learning based optimization algorithm based on course by course improvement. In: IEEE. ; 2015: 48–52.

27. Rao RV, Savsani V, Balic J. Teaching–learning-based optimization algorithm for unconstrained and constrained real-parameter optimization problems. *Engineering Optimization* 2012; 44(12): 1447–1462.

28. Rao RV, Patel V. An improved teaching-learning-based optimization algorithm for solving unconstrained optimization problems. *Scientia Iranica* 2013; 20(3): 710–720.

29. Niu P, Ma Y, Yan S. A modified teaching–learning-based optimization algorithm for numerical function optimization. *International Journal of Machine Learning and Cybernetics* 2019; 10(6): 1357–1371.

30. Wu HC. The Karush–Kuhn–Tucker optimality conditions in an optimization problem with interval-valued objective function. *European Journal of Operational Research* 2007; 176(1): 46–59.

31. Tseng CC, Lee SL. Closed-form design of fractional delay FIR filter using discrete Hartley transform. In: IEEE. ; 2007: 1–4.

32. Tseng CC, Lee SL. Design of fractional delay filter using discrete Fourier transform interpolation method. *Signal Processing* 2010; 90(4): 1313–1322.

33. Tseng CC, Lee SL. Closed-form design of fractional order differentiator using discrete cosine transform. In: IEEE. ; 2013: 2609–2612.

34. Tseng CC, Lee SL. Design of fractional delay FIR filter using radial basis function. In: IEEE. ; 2009: 485–488.

35. Srivastava S DA, D N. Optimal design of symmetric fractional delay filter using firefly algorithm. *International journal of circuit, theory and application* 2020: 1–12.