ISSN 2229-5518

A Review on Nature Inspired Computation based Adaptive Channel Equalization

Annapurna H S, Dr. A. Rijuvana Begum

Abstract—In the digital communication channel, the transmitted signal might get dispersive due to which the information may not be carried as the same is transmitted. Additive noise and inter-symbol interference (ISI) cause dispersal of the signal. Channel equalization is a key method used in the digital communication system. This mitigates the result of inter-symbol inference in disruptive channels. The ISI can be curtailed if the channel is specifically known. To recompense the intrinsic residual distortion equalization used. A comprehensive study of state-of-the-art study on nature-inspired computation approaches incorporated in channel equalization is presented. Several methods employed in adaptive channel equalization are debated in the literature. Adaptive algorithms used as optimization techniques are defined. Some of the technique that has been presented is Particle Swarm Optimization, Genetic Algorithm, Functional Link Artificial Neural Network, and Neural Network.

Index Terms— Adaptive algorithms, Channel equalization, Digital communication channel, Functional Link Artificial Neural Network, Genetic Algorithm, Inter symbol interference, Particle Swarm Optimization,

1 Introduction

THE digital signal communication got a vital range of applications as the digitalization arrived. These types of applications lead digital province several modulation systems and more updates. Even though so many schemes and their updates exist, they have strappingly effected two basic problems that arise in conventional digital communication schemes. The two major problems have more impact are noise and Inter-Symbol-Interference (ISI). The reasons for this kind of error generating phenomenon are channel features associated between transmitter and receiver and the spreading of the transmitted signal. The noise outcome on communication hinges on the channel characteristics and it can be condensed with appropriate channel selection. Even the channel is found too noisy the signal received can be less affected if the SNR sustained at the receiver by improving transmitting signal strength. ISI on symbol energy is spread into another symbol duration which scatters the symbol and effects the communication [1].

The prevailing adaptive equalizers that employ linear combiner structure using the LMS algorithm and its multiple variants accomplish inadequately in the existence of nonlinear distortions, as it cannot converge to the global minimum solution for the multimodal and non-uniform objective function. Some of the few general techniques among them are GA, BFO, AIS, DE, PSO, and its variants [2]. To alleviate the effect of ISI, the first equalizer structure was proposed. Generally, the coefficients of adaptive linear combiner are adjusted with gradient descent systems such as LMS, normalized LMS (NLMS), recursive least square (RLS), etc. These linear FIR equalizers provide better performance for linear channels. However, most of the practical wireless channels are strictly non-linear owing to the manifestation of non-linearity in data converters. Moreover, in satellite communication, amplifier saturation in satellite also contributes to non-linearity. Hence, there is a need for a non-linear model to recover the data transmitted through the non-linear channels. Neural network [NN] based equalizers are the potential solution to combat this type of nonlinear alteration in the wireless communication channel. NN based equalizer offers a faster convergence rate with lesser Mean Squared Error and BER due to nonlinear signal processing capability. Due to non-linear signal processing capability, neural networks (NNs) can create arbitrary complex decision regions. Due to these features, networks with different structures are successfully applied in channel equalization. To reduce complexity due to multiple layers, some single layer NN structures such as radial basis function (RBF), polynomial perceptron network (PPN), FLANN, Chebyshev neural network (ChNN) has been used for equalization [3]. To handle the time-varying phenomenon of fading channels, the use of Recurrent Neural Network was proposed. The RNN structure comprises an IIR filter with a feedback system, along with nonlinearities associated with the neurons. The RNN outperforms the MLP and RBF structures and could be used for both trained as well as blind equalization [4].

The rest of the research paper is arranged as follows. In Section II, several studies related to channel equalization is discussed. In section III, various nature-inspired computation techniques employed for channel equalization is mentioned. Finally, in section IV, the conclusion of the survey is presented.

2 Literature Review

Wireless communication channels were increasingly being pushed into the paradigm nonlinearity and ISI because of the demand for high-speed data through portable and powerefficient handheld devices. ISI and nonlinearity cause severe deprivation in established signals resulting in a humble quality of service. The channel equalizers were planned for justifying channel nonlinearity and ISI.

An enhanced method of training wavelet neural network-based equalizer by PSO was proposed which comprises optimizing the translation, dilation, and other weights of hidden layers to attain enhancement in BER performance [5]. RBF-based NN equalizers are a striking alternate and have efficaciously been applied for blind equalization. RNN-based equalizers, generalized as IIR filters, outperform feed-forward NNs, comprising MLP, RBF, and FLANN [6].

ONS Space-time processing is a rapidly emerging field that shows significant promise in improving the enactment of communication networks. Blind ST processing seems to be powerful leverage for the improvement of performance [7]. Former methodologies were based on the minimization of the MSE performed by a gradient descent procedure. The MSE is not necessarily related to the classification error—bit error rate (BER)—that is considered in equalization problems; moreover, the use of gradient-based learning techniques is often hampered by the slow speed convergence and statistical ill-conditioning. Experimental tests conducted on 2-PAM signals for dissimilar channels have established a better performance of the novel algorithm [8].

Adaptive algorithms such as Least-Mean-Square (LMS) based channel equalizer intent to lessen the Intersymbol Interference (ISI) existing in the transmission channel. However, adaptive algorithms agonize from long training periods and detrimental local minima throughout training mode. A novel adaptive channel equalizer utilizing the Genetic Algorithm which is a derivative-free optimization tool is discussed. The recital of the proposed channel equalizer is estimated by MSE, convergence rate, and is related to its LMS and RLS counterparts. It is perceived that the new adaptive equalizer based GA offers a better performance so far as the accuracy of reception is fretful [9].

A rationalized structure for adaptive filters has been presented in which the major adaptive filter algorithms: LMS, NLMS, APA, RLS, TDAF, and PRSAF are all easily derived in a unified way. The differences between the various algorithms are identified as differences in the selection of a pre-conditioner. A further benefit of the streamlined approach is the possibility of doing performance analysis on general recursions rather than for each algorithm [10]. An ideal step-size finding algorithm for LMS is existing that runs iteratively and convergence to the equalizer coefficients by finding the optimal step-size which diminishes the steady-state error rate at all iteration. No initialization for the step-size value is required. The ability of the anticipated algorithm is presented by constructing a performance comparison among specific additional LMS based algorithms and optimal step-size LMS algorithm [11]. A complex stochastic gradient adaptation was proposed so that the MPNN equalizer can adapt to slowly varying channels. Simulations have shown that the network was able to self-adjust the position of the centers and the weights associated with each center to the changing channels. The method suggested uses measurements of the Euclidean distance of one center to the other centers. Simulations showed that for certain channels, the computational savings during runtime by reducing the size of the network could be enormous [12]. The problem of adaptive equalization was explored when the channel diversity condition is not satisfied.

A prediction error equalizer with a RELS algorithm was proposed that showed the adaptive equalizer is universally stable, the parameter evaluations were consistent, and the prediction error congregates toward a scalar multiple of the input sequence [13]. Signed Regressor FLANN based non-linear channel equalizer has been employed over complex-valued non-linear channels and the results showed that such an algorithm was capable of constructing a simple network whose performance is adjacent to the optimal solution [14]. Quite a few NN structures and learning methods to address the issues of channel equalization were discussed by considering several representative NN algorithms. Furthermore, two ML-based equalizers, DNN- and CNN-based equalizers, were presented, and their performance was examined and compared with two predictable equalizers developed based on the LS and MMSE principles. The results showed that unlike outmoded channel equalization and recognition methods, no channel statistics were prerequisite to be discretely computed in ML-based equalization methods. More prominently, the BER performance of OFDM systems engaging ML-based equalizers was found to be classically superior to that of OFDM systems employing conventional LS- and MMSE-based equalizers. The nature-inspired computation-based equalizers also have a higher capability to learn and analyze complicated properties of wireless channels and are also more effective to combat ISI [15].

3 NATURE INSPIRED COMPUTATION TECHNIQUES

Adaptive algorithms used as optimization techniques are described below as follows.

3.1 Particle Swarm Optimization

PSO algorithm suggestively hinges on group behavior of birds, etc., with their natural habits of grouping. In PSO, an abundant particle proliferates everywhere in solution space to find out the ultimate solution. They also stare at the best elements in their path. So particles think about their individual finest solutions with remaining solutions lay down so far. PSO was precisely modeled as follow:

$$velc(t+1) = w.velc(t) + p_1.rand.(pbest(t) - posc(t)) + p_2.rand.(gbest(t) - p$$

$$posc(t+1) = posc(t) + velc(t+1)$$

Where t, velc(t), posc(t), pbest(t) are the prevailing number of iterations, the velocity of particles, position of particles, and personal finest position of the particle, respectively. Also, gbest(t) is the global best position, and p1, p2 are acceleration coefficients [16].

3.2 Firefly Algorithm (FFA)

The firefly flashing is an extraordinary sight in tropical and temperate areas. The critical characteristic of sparks is to engaged mating partners and entreat potential prey. Basic concepts for FFA are: (1) a firefly may be affianced to another despite their group. (2) For a couple of blinking fireflies, lighter fireflies will follow a brighter one. But if brighter firefly is not available, then it travels erratically [16]. For global minima, brilliance can only inversely proportionate to the approximation of a working function. Firefly algorithm is focused on two crucial notions, one is an incongruity in light intensity and the other is the commencement of attraction of other fireflies. The desirability of a firefly is dogged by its brightness which is by some means associated with working function. The brightness of the firefly at fixed spot x is chosen as the G(x). For any certain medium, light absorption coefficient γ, light intensity G varies with distance r as,

$$G = G_0 e^{-\gamma r^2}$$

Here G0 is the actual radiant intensity at r = 0. The distance between two fireflies i and j can be specified as,

Adaptive Channel Equalization Using Decision Directed …

$$r_{ij} = \|x_i - x_j\| = \sqrt{\sum_{k=1}^{D} (x_{i,k} - x_{j,k})^2}$$

Where

x_i, k is *k*th element of *i*th firefly.

The indication of *i*th firefly is employed to one more prominent *j*th firefly and is estimated by,

$$x_i = x_i + \beta_0 e^{-\gamma r_{ij}^2}(x_j - x_i) + \alpha(rand - \frac{1}{2}); \beta_0; is + ve$$

3.3 Ant Colony Optimization

Ant Colony Optimization (ACO) is a model for designing meta-heuristic algorithms and resolving solid combinatorial optimization hitches stimulated by the indirect communication of real ants. In ACO algorithms, ants build candidate resolutions to the delinquent being tackled, making decisions that are stochastically biased by numerical information based on (artificial) pheromone trails and available heuristic information. The pheromone tracks are rationalized throughout algorithm accomplishment to bias the ants pursuit toward promising decisions formerly found.

Foraging: In foraging Individual ants bonds a chemical on the ground which raises the probability that other ants will follow the same path. Biologists have publicized that numerous colony-level behaviors detected in social insects can be elucidated via rather modest models in which only stigmergic communication is present. Dissimilar phases of the comportment of ant colonies have encouraged various types of ant colony algorithms. Division of labor, brood sorting, and cooperative conveyance are the best examples of foraging.

Division of labor: Division of labor is an imperative and extensive feature of life in ant colonies. Social insects are all characterized into two types i.e., fundamental type of division of labor & reproductive division of labor. There are diverse types of division of labor i.e., reproductive, castes, tasks proficient in the colony.

Collective transport: Collective behavior grosses many forms, such as advent, self–organization, superorganism, quorum sensing, artificial intelligence, and dynamic networks.

Cluster formation: Clustering problems that are mostly inspired by the behavior of ant colonies and this behavior based upon the brood sorting. In Brood sorting primarily ant colonies sort their brood in the method of smallest items in the middle and largest items in the edge. The best example of brood sorting is communal structure formation by social insects. By the comportment of ant colonies, they cluster their corpses and sort their larvae. The ant colony clustering algorithm is deployed to improve cluster benchmark problems [18].

3.4 Artificial Neural Network

An ANN takes the name from the system of nerve cells in the brain. Recently, ANN is an imperative technique for classification and optimization problems. NNs have been widely used in numerous signal processing applications. Feed-forward neural network based on backpropagation algorithm, linear layer(train), layer recurrent & NARX network type that exploits the principle of discriminative learning, by minimizing an error function is being reconnoitered. The performance of suggested approaches can also be associated with adaptive equalizer based on the LMS algorithm. An LMS equalizer using a feed-forward neural network is built on a back-propagation algorithm & compared its performance with adaptive equalizer based on a neural network. The main section of the back-propagation algorithm is the high speed of convergence concerning gradient-based approaches. Simulation regarding the equalization of QAM signals in the AWGN transmission channel is described, which demonstrated the usefulness of the recommended technique [17].

3.5 The Genetic Algorithm

GAs is evolutionary techniques that use a Darwinian criterion of population evolution generally called stochastic search mechanism. This procedure of natural selection is utilized to raise the effectiveness of a group of probable solutions to obtain an environmental optimum [17]. The most common algorithm is Gradient-descent training used in signal processing today because they have a solid mathematical basis however gradient-descent training has few limitations:

· Derivative built algorithm so there are chances that the fac-

ISSN 2229-5518

tors may fall to local minima conditions during training.

· Do not perform reasonably under high noise surroundings and for nonlinear channels

· In certain cases, they do not accomplish reasonably if the order of the channel rises

· LMS algorithm at times shows slower convergence

These restrictions can be detached by using evolutionary algorithms such as genetic algorithms. A genetic algorithm uses the progression of natural choice and does not involve error gradient statistics. A GA can find global error minima for any given problem.

Classically, a genetic algorithm consists of the following steps.

- Initialization-an initial population of the search nodes is randomly generated.
- Evaluation of the fitness function- The fitness value of each node is calculated allowing the fitness function (objective function).
- Genetic operations-new search nodes are generated randomly by examining the fitness value of the search nodes and applying the genetic operators to the search nodes
- Repeat steps 2 and 3 until the algorithm converges.

3.6 Fuzzy Neural Networks

The kernel of a fuzzy inference system is a fuzzy knowledge base. In a fuzzy knowledge base, the info that comprises of input-output data points of the system is inferred into linguistic interpretable fuzzy rules. The fuzzy rules that have IF-THEN form and erected by using nonlinear quadratic functions are used. They have the following form.

If is and…and is Then $y_j = \sum_{i=1}^{m} (w1_{ij} x_i^2 + w2_{ij} x_i) + b_j$

Here $x_1, x_2, \ldots, x_m$ are input variables, y_j =(j=1,…,n) are output variables which are nonlinear quadratic functions, A_{ji} is a membership function for i-th rule of the j-th input defined as a Gaussian membership function. $w1_{ij}$, $w2_{ij}$ and b_j (i=1,..m, j=1,…,n) are parameters of the network. The fuzzy model that is described by IF-THEN rules can be attained by amending constraints of the conclusion and premise parts of the rules. A gradient method is used to train the parameters of rules in the neuro-fuzzy network structure. By using fuzzy rules in the equation, the structure of the NNFN is proposed. The NNFN comprises seven layers. In the first layer, the number of nodes is equal to the number of input signals. These nodes are deployed for allocating input signals. In the second layer, each node corresponds to one linguistic term. For each input signal inward bound the system, the relationship degree to which input value belongs to a fuzzy set is considered. To define linguistic terms the Gaussian function is used.

A DFNN is being pragmatic to the problem of digital communication channel equalization problems for the last few years. Over combining the neural network learning proficiencies and fuzzy rules, DFNN avail the advantages of both the fuzzy logic and neural networks. DFNN equalizer is grander to other equalizers such as RNN and minimum resource allocation network (MRAN) in terms of MSE and BER [20].

3.7 Channel Equalization using Deep Learning

The pilot-only channel estimation as a multivariate regression problem is taken. The input is the binary array Y˜t while the output is the complex-valued Hˆ. Multivariate regression is a hard problem and many times it is tackled by grouping N multivariable regressors, which are trained independently for simplicity. This method adopted a single neural network (NN) to perform multivariate regression. It should be noted that the literature on NNs is very rich and precedes DL. For example, NNs for channel equalization was used in many other communication problems. Besides, for non-perceptual data or when data is scarce, there are algorithms such as gradient boosting that are highly competitive with DL.

The NN has trained with the mean-squared error (MSE) loss and aims at providing the minimum MSE (MMSE) estimation.

$$H^{\dagger} = \min_{\hat{H}} \Xi\left[\| H - \hat{H} \|^2\right]$$

The 1-bit quantization means that it is not possible to estimate the norm of the channel with zero-threshold quantizers. This information can be recovered from the automatic gain control in the analog circuitry. Therefore, we suppose that both training and test data have NtNr and normalize the channel after the MMSE estimate.

The training does not require knowledge of the distribution p(H) over channels, but access to a reasonable number of realizations (to compose a rich training set from e. g. measurement data) or a software routine to draw samples from this distribution on-the-fly. In contrast, state-of-art GAMP-based algorithms consider the receiver knows the distribution p(H) of channels but not its realizations. Knowing distributions for AMP (even if not their parameters) and having large datasets for DL, are similar in the sense that both are manifestations of access to a potentially infinite amount of data.

One distinction is that GAMP variants leverage the analytical expression of p(H) as a highly compact representation of knowledge about the channels. When trained, the NN is expected to find its way of representing all relevant information contained in p(H).

Similarly, they train a NN under different noise conditions (multi-condition training) and expect it to learn and generalize on the conditions of interest. Both noise multi-condition training and the time-variant channel are challenging for the Stochastic Gradient Descent used in DL. The network training with SGD may not converge even with advanced Keras' optimizers such as Adam. Most SGD routines obtain the gradient estimate by averaging the individual gradients of a set of B examples called mini batch. Having B > 1 often helps convergence by averaging the noise out and maybe essential when the SNR imposed during training is low. In the case of time-varying channels, SGD may not find a reasonable average direction even if σ2 is small [21].

3.8 Recurrent Neural Network

RNN is generous of neural network and its networks among nodes form a directed graph laterally a temporal structure. Contrasting to feed-forward neural networks, RNN could use the internal state to practice sequences of inputs. Conferring to the above-mentioned features, premeditated the joint channel equalization and decoding neural model based on the structure of RNN. A Gated Recurrent Unit (GRU) was suggested to deal with the exploding and vanishing gradient difficulties for the RNN with less training parameters. The GRU has reset and update gates which can control the data flow and parameter passing. The bi-directional Gated Recurrent Unit (bi-GRU) connects two hidden layers of conflicting directions to the same output, which can increase the performance of GRU [19].

3.9 Wind Driven Optimization

WDO is a population-based iterative heuristic global optimization technique similar to other nature-inspired optimization algorithms, aiming to improve the best candidate solution over time. It is highly correlated with the actual physical equations describing the trajectory of an air parcel in our atmosphere. The populations of air parcels are ranked in sliding order based on their pressure values such that its new velocity unew and new position xnew can be represented as,

$$u_{new} = (1-\alpha)u_{cur} - g x_{cur} + \left(RT \left| \frac{1}{i-1} \right| \left(x_{opt} - x_{cur} \right) \right) + \frac{c u_{cur}^{other\ dim}}{r}$$

$$x_{new} = x_{cur} + u_{new}\Delta t$$

where u_{cur} is the velocity at the current iteration, α and g being friction coefficient and gravitational constant respectively, x_{cur} is the current location, RT defines universal gas constant and temperature, C is the Coriolis force, x_{opt} being the optimum location, r is the ranking among all parcels and $cu_{cur}^{otherdim}$ is the replaced velocity vector from another randomly chosen dimension to represent the influence of Coriolis force. A time step, Δt equal to 1 is assumed. Coriolis force and gravitational pull in WDO provide a favorable contribution which prevents air parcels from remaining trapped at the boundary for a long period and pulls them back into the search space. WDO coefficients can be fine-tuned for different optimization topologies to provide potential benefits. The air parcel population is ranked centered on their pressure value (cost function) and velocity is updated with the following limitation,

$$u_{new}^{*} = \begin{Bmatrix} u_{\max} if u_{new} > u_{\max} \\ -u_{\max} f u_{new} < -u_{\max} \end{Bmatrix}$$

Where, the path of motion is conserved but the scale is limited to |umax| at any dimension and u*new represents the adjusted velocity after it is limited to the maximum speed. The lowest-ranked fitness function is taken as xopt value [22].

3.10 Training Weights of the Equalizers with Moth Flame Optimization

Moths are kind of insects that are analogous to the family of butterflies and can navigate by using transverse orientation. In this movement, they uphold a fixed angle to the moon, which is quite far and hence travel in a straight line. However, when uncovered to the simulated lights which are nearer, they move spirally and locate their best positions till then. In this algorithm, the candidate resolutions are the moths (M) and the problem variables are the flames (F). The process has been used for channel equalization and the step-by-step procedure is described below:

1. Initialization of moths and flames:

Let N be the number of moths and flames each having Q number of features which hinge on on the order of the equalizer to be optimized. For example, if the equalizer has two first-order factors in numerator and two second-order factors in denominator, the order of the equalizer is four but the number of parameters to be leveled is seven (sum of a's and b's) as can be depicted by the following equation.

$$H(k) = K \frac{(a_0 + a_1 z^{-1})(1 + a_2 z^{-1})}{(1 + b_0 z^{-1} + b_1 z^{-2})(1 + b_2 z^{-1} + b_3 z^{-2})}$$

2. Evaluation of fitness function:

The outcome of the channel laterally with the added AWGN noise is passed as input to the equalizer and the output of equalizer is found out. Formerly, the fitness function is assessed for all the N candidates. For FIR channel, fitness is mean square error and for IIR channel, it is JCM.

3. Sorting:

Sort the moths and their fitness factor values to minimize the fitness function.

4. Updation of weights:

Update the moth positions conferring to the following equation:

$$M_i = F_j + \delta_i . e^{kt} . \cos(2\pi t)$$

$$\delta_i = \left| F_j - M_i \right|$$

The k is a constant for outlining the shape of a logarithm spiral (defining the crusade of the moth towards the flame) and t is a random number between ½r; 1, where r is linearly decreased from - 1 to - 2, to highlight exploitation. This means a lesser value of t conforming to a closer position to the flame as compared to a higher value of t.

5. Update the total number of flames:

The flame number is updated using,

$$Flameno = round\left(N - itr.\frac{N-1}{T} \right)$$

Where T is the total amount of iterations and itr is the current iteration.

6. Update the location of the flame:

Equate the earlier flame position with the existing moth positions. If the current moth position is found to be better, then update the flame position to this current moth position. Also, update the best flame position if a new better value is found.

7. Stopping Criteria:

Either the desired fitness factor is achieved or the number of iterations reaches to the maximum defined limit. Once it is achieved, the optimized weights of the equalizer are obtained [23].

4 CONCLUSION

Several nature-inspired computation methods have been discussed for the adaptive channel equalization problem. GA, PSO was used to form NN, but these are restricted in updating the ANN weight. Again, PSO research is limited to space and can easily fall into a local minimum. WDO aims to improve the best candidate solution over time and it is highly correlated with the actual physical equations describing the trajectory of an air parcel in our atmosphere. DFNN equalizer is superior to other equalizers by comparing BER. Vector Support Machines (SVMs) can be used to equalize the non-linear recursive channels as future work.

REFERENCES

[1] Varma, D. S., Kanvitha, P., & Subhashini, K. R. (2019, April). Adaptive Channel Equalization using Teaching Learning based Optimization. In *2019 International Conference on Communication and Signal Processing (ICCSP)* (pp. 0001-0006). IEEE.

[2] Sinha, R., Choubey, A., Mahto, S. K., & Ranjan, P. (2019). Quantum Behaved Particle Swarm Optimization Technique Applied to FIR-Based Linear and Nonlinear Channel Equalizer. In *Advances in Computer Communication and Computational Sciences* (pp. 37-50). Springer, Singapore.

[3] Ingle, K. K., & Jatoth, R. K. (2020). An Efficient JAYA Algorithm with Lévy Flight for Non-linear Channel Equalization. *Expert Systems with Applications*, *145*, 112970.

[4] Nanda, S. J., & Jonwal, N. (2017). Robust nonlinear channel equalization using WNN trained by symbiotic organism search algorithm. *Applied Soft Computing*, *57*, 197-209.

[5] Majumder, S., & Giri, M. K. (2020). Nonlinear Channel Equalization Using Wavelet Neural Network Trained Using PSO. *Available at SSRN 3572806*.

[6] Burse, K., Yadav, R. N., & Shrivastava, S. C. (2010). Channel equalization using neural networks: A review. *IEEE transactions on systems, man, and cybernetics, Part C (Applications and Reviews)*, *40*(3), 352-357.

[7] Papadias, C. B., & Paulraj, A. J. (1997, April). Space-time signal processing for wireless communications: a survey. In *First IEEE Signal Processing Workshop on Signal Processing Advances in Wireless Communications* (pp. 285-288). IEEE.

[8] Parisi, R., Di Claudio, E. D., Orlandi, G., & Rao, B. D. (1997). Fast adaptive digital equalization by recurrent neural networks. *IEEE Transactions on Signal Processing*, *45*(11), 2731-2739.

[9] Mohammed, J. (2012). A study on the suitability of genetic algorithm for adaptive channel equalization. *International journal of electrical and computer engineering*, *2*(3), 285.

[10] Husøy, J. H., & Abadi, M. S. E. (2008). Unified approach to adaptive filters and their performance. *IET signal processing*, *2*(2), 97-109.

[11] Solmaz, C. Ö., Oruç, Ö., & Kayran, A. H. (2011, April). Optimal step-size LMS equalizer algorithm. In *2011 IEEE 19th Signal Processing and Communications Applications Conference (SIU)* (pp. 853-856). IEEE.

[12] Padhy, S. K., Panigrahi, S. P., Patra, P. K., & Nayak, S. K. (2009). Non-linear channel equalization using adaptive MPNN. *Applied Soft Computing*, *9*(3), 1016-1022.

[13] Radenkovic, M. S., & Bose, T. (2009). A recursive blind adaptive equalizer for IIR channels with common zeros. *Circuits, Systems & Signal Processing*, *28*(3), 467-486.

[14] Dash, S., Sahoo, S. K., & Mohanty, M. N. (2013). Design of adaptive FLANN based model for non-linear channel equalization. In *Proceedings of the Third International Conference on Trends in Information, Telecommunication and Computing* (pp. 317-324). Springer, New York, NY.

[15] Zhang, L., & Yang, L. L. (2020). Machine Learning for Joint Channel Equalization and Signal Detection. *Machine Learning for Future Wireless Communications*, 213-241.

[16] Sarangi, A., Sarangi, S. K., & Panigrahi, S. P. (2018). Adaptive Channel Equalization Using Decision Directed and Dispersion Minimizing Equalizers Trained by Variable Step Size Firefly Algorithm. In *Intelligent Engineering Informatics* (pp. 301-310). Springer, Singapore.

[17] Kundu, D., & Nijhawan, G. (2017). Performance Analysis of Adaptive Channel Equalizer Using LMS, Various Architecture of ANN and GA. *International Journal of Applied Engineering Research*, *12*(22), 12682-12692.

[18] Himabindu, K., & Jyothi, J. (2017). Nature inspired computation techniques and its applications in soft computing: survey. *Int J Res Appl Sci Eng Tech*, *5*, 1906-1916.

[19] Hu, Y., Zhao, L., & Hu, Y. (2019, June). Joint channel equalization and decoding with one recurrent neural network. In *2019 IEEE International Symposium on Broadband Multimedia Systems and Broadcasting (BMSB)* (pp. 1-4). IEEE.

[20] Kaur, G., & Kaur, G. (2019). Non-linearities mitigation with fuzzy neural networks using a machine learning algorithm in a CO-OFDM system. *IET Optoelectronics*, *14*(1), 44-51.

[21] Cheng, X., Liu, D., Wang, C., Yan, S., & Zhu, Z. (2019). Deep learning-based channel estimation and equalization scheme for FBMC/OQAM systems. *IEEE Wireless Communications Letters*, *8*(3), 881-884.

[22] Sinha, R., & Choubey, A. (2017). Soft Computing Techniques to Estimate FIR Filter Weights in an Adaptive Channel Equalizer: A Comparative Study. *International Journal of Applied Engineering Research*, *12*(13), 3988-3995.

[23] Nanda, S. J., & Garg, S. (2019). Design of Supervised and Blind Channel Equalizer Based on Moth-Flame Optimization. *Journal of The Institution of Engineers (India): Series B*, *100*(2), 105-115.