Hybrid Artificial Neural Network by Using Differential Search Algorithm for Solving Power Flow Problem

Kadir ABACI, Volkan YAMAÇLI
Department of Electrical-Electronics Engineering, Mersin University, Mersin, 33343, Turkey
vyamacli@mersin.edu.tr

***Abstract*—Power flow (PF) is in one of the most studied non-linear problems related to power systems which heavily affects security issues such as generation cost, voltage stability and active power loss. In this paper, a simple and new approach based on artificial neural network (ANN) and differential search (DSA) algorithm has been proposed and applied for one of the most complex problems in power systems, Power Flow (PF) problem. By using the proposed DSA implemented ANN method, IEEE 9-bus, IEEE 30-bus and IEEE 118-bus test system parameters are obtained without running iterative convergence methods such as Gauss-Siedel or Newton-Raphson. By comparing with several most used non-linear iterative methods, the results obtained using the classical training method and proposed DSA implemented hybrid training methods are presented and discussed. Obtained results in this work show that the ANN based power flow method can be implemented to solve non-linear static and dynamical problems concerning power systems successfully.**

***Index Terms*—heuristic algorithms, iterative methods, neural networks, optimization, power system analysis computing.**

I. Introduction

In general, power flow (PF) is defined as a non convex, non-linear and multi-dimensional problem depending on power system data. PF becomes more complicated due to system constraints whereas satisfying the system parameters. Power system operating is a challenging task where modern electrical systems must be able to fulfill for the continually changing load demand while providing high quality energy [1]. In addition to this, in order to optimize power system output such as generation fuel cost minimization or voltage profile improvement, which are user-defined optimizations cost functions; optimal power flow is used and applied. Since a power system is modeled as set of constants and derivative variables, linear methods cannot easily converge to a solution ensuring desired and minimum deviation. Due to increasing energy demand and expanding distribution networks, the control and stabilization of power systems become difficult because of unscheduled power outages and voltage drops. The main objective of PF is converging to best system parameters such as active and reactive power generation, injected reactive power values, load tap changer ratios and busbar voltages while compensating the system constraints in order to secure system stability and prevent system faults. PF solution determines the best control and parameter variables for the most efficient power system planning and operation [2].

In literature, various PF algorithms based on different classical numerical methods are used and studied. Following the first successful implementation of these methods is reduced gradient method, proposed by Carpentier [3] as some other methods are also used with their advantages and disadvantages. Linear programming (LP) have been implemented and used in [4, 5] by virtue of ability to converge fast and better results compared to other methods whereas quadratic programming (QP) [6] is also used in order to determine system parameters. In addition to this, Newton-Raphson iterative method [7, 8] is also used in power flow and optimal power flow problems. Sequential unconstrained minimization technique (SUMT) [9] and interior point method [10] are also used to solve power flow problems. In order to use numerical methods, it is required to determine and define an initial point for considered solution set which needs to be close to global minimum point due to possible of being stuck on local minimums or not being able to converge in different constraints. In last decades, thanks to the development of computing and data processing technologies, using new methods such as neural networks; deep learning; heuristic methods and clustering based optimization techniques in order to solve power flow has grown, rapidly. Some of these methods can be applied to system without having trained whereas some other methods should be trained with sample data in order to determine system parameters which are mostly constants and coefficients.

Artificial neural network (ANN) is a nature-inspired method consists of artificial neurons which process input data with suitable coefficients as representation of mathematical weights and constants. ANNs can be defined as layers of exponential functions set which are designed to connect each other whose aim is acquiring system output without simulating the system or calculating actual system variables by weighting the input with desired coefficients and employing various bias functions. In most studies, ANN is used to determine most satisfying system parameters for a given power system. ANNs have been studied as a new approach for data processing with an algorithmic procedure for solving a problem due to being capable of producing an output if an appropriate data is provided about the problem and are referred to as learning or adaptive models [11]. In order to train an ANN, there are various methods and algorithms which use different approaches and functions.

This work was supported by Mersin University Scientific Research Council: Project 2018-1-TP3-2868.

Digital Object Identifier 10.4316/AECE.2019.04007

These functions can be numeric, iterative or heuristic depending on complexity of system and topology of the network. Thus, in some studies nature-inspired heuristic optimization algorithms which use random iterative rules are also applied as a training function in order to optimize the network for ensuring stability and convergence.

Karaboga et al. presented and proposed a new training method by using one of the most popular heuristic algorithms, artificial bee colony algorithm (ABC) in different studies for signal processing [12] and pattern classification [13]. In [14] ABC is also used in order to compare with back-propagation training algorithm that heuristic algorithm produce better results than classical training method. Genetic algorithm (GE) is used in order to train ANN for better training results in studies related to store forecasting [15] and ultrasonic gas flowmeter [16] where it is also seen that heuristic methods outputs better network parameters compared to other iterative training methods. In literature, there are also recent studies on training ANNs by using ant colony algorithm [AC] for different objective functions and complex systems [17, 18].

Lately, with the increasing interest in heuristic and population based optimization techniques, a new and efficient algorithm based on brownian-like random-walking behavior, differential search algorithm (DSA) is presented [19]. The DSA uses migration behavior of super-organisms in order to achieve the best and optimal result in given problem.

In literature, DSA is used in power system related problems [1, 20] as well as image processing studies such as multilevel color image tresholding [21] and edge detection [22] while also used in papers focused on optimizing fuzzy and PID controller parameters [23, 24]. Despite being used for various optimization problems, no other study in literature is reported for using DSA in order to train and optimize ANN coefficients. In this paper, a novel ANN based approach by using hybrid DSA training method is proposed with the purpose of solving PF problem. In order to test and show the efficiency of proposed method, IEEE 9-bus, IEEE 30-bus and IEEE 118-bus test systems are used and studied.

II. Power Flow

Power flow problem, also known as load flow, is one of the most studied non-linear problems in power systems [20]. In order to achieve the best result, system variables are chosen initially whereas iterative method is applied to determine the best system parameters. The PF problem handled in this study aims at converging to the best results for single objective non-linear problem such as swing bus power.

Power flow problem can be defined as:

Solve: *f(x,u)*

By using: *g(x,u)= 0* and *h(x,u)<0*

f and *g* symbolizes the objective function and the load flow equations as h shows the parameter limit.

$$x = \left[P_{Gslack}\, V_L\, Q_G\, S_l \right] \tag{1}$$

The parameter x shows the variables of slack bus power generation, load bus voltages, reactive power generation and line load.

$$u = \left[P_G\, V_G\, Q_C\, T \right] \tag{2}$$

u indicates system matrix variables including generator power, voltage, injected reactive power and tap changer transformers ratio.

Power related equations are also given by (3, 4),

$$P_{Gi} - P_{Di} - \sum_{j=1}^{n} |V_i||V_j||Y_{ij}|\cos(\theta_{ij} - \delta_i + \delta_j) = 0 \tag{3}$$

$$Q_{Gi} - Q_{Di} - \sum_{j=1}^{n} |V_i||V_j||Y_{ij}|\sin(\theta_{ij} - \delta_i + \delta_j) = 0 \tag{4}$$

the bus admittance matrix elements are represented by $|Y_{ij}|$ and θ_{ij} with indicating n as the total bus number.

A. System Constraints

Voltage magnitudes, active and reactive power limits of system generators including slack bus are given by (5-7)

$$V_{Gi}^{min} \le V_{Gi} \le V_{Gi}^{max} \quad i = 1, \dots, N_g \tag{5}$$

$$P_{Gi}^{min} \le P_{Gi} \le P_{Gi}^{max} \quad i = 1, \dots, N_g \tag{6}$$

$$Q_{Gi}^{min} \le Q_{Gi} \le Q_{Gi}^{max} \quad i = 1, \dots, N_g \tag{7}$$

Tap ratio settings regarding the transformers are given by,

$$T_i^{min} \le T_i \le T_i^{max} \quad i = 1, \dots, N_T \tag{8}$$

Upper and lower limit of injected reactive power is defined as,

$$Q_{Ci}^{min} \le Q_{Ci} \le Q_{Ci}^{max} \quad i = 1, \dots, N_{QC} \tag{9}$$

Transmission line apparent power and loaded bus voltage values are also need to be within constraints given by the user,

$$V_{Li}^{min} \le V_{Li} \le V_{Li}^{max} \quad i = 1, \dots, N_{PQ} \tag{10}$$

$$S_{Li} \le S_{Li}^{max} \quad i = 1, \dots, N_L \tag{11}$$

III. Artificial Neural Network Model

ANN, namely neural network (NN) is the mathematical model based on neural system of living organisms such as humans or animals. Despite having similar topologies for different problems, ANNs have a non-linear link between the input variables and output data which is created by using coefficients and bias factors depending on the system [25]. By using ANNs various real-world problems such as complex functional approximation problems, pattern classification, clustering and image processing can be solved and the output is determined accurately if the system is trained with sufficient data. [26]. An ANN consists of an input layer and an output layer basically. Also there are hidden layers between input and output layers which are used to estimate to desired output by determining hidden parameters according to non-linearity of problem in question.

ANNs can be designed depending on the problem including different number of input and output variables. But in order to work on multiple input and multiple output systems, the ANN should be designed similar to a surjective function that there should be accurate output(s) regardless of variation of input values. The number of layers, neurons per layers, training algorithms and validation values should be chosen appropriately considering network robustness and training time should.

In order to use ANN to estimate and determine a system output within desired performance, the network should be trained with sufficient samples known as training dataset depending on the number of system parameters, complexity and non-linearity of the system. Since the aim of this study is obtaining a simple robust multilayer feed-forward ANN model, a core topology is used and applied which is shown in Figure 1. ANN training algorithm determines the decision point updates the coefficients of the network. There are many training algorithms used in literature. [26]. In feed-forward ANN model a node receives a signal from the previous layer and multiplies the mathematical signal with a coefficient. The optimization goal is to minimize the objective function by optimizing the network weights. In this study mean square error (MSE) is chosen as the output error function in order to determine and converge to the best network weights.

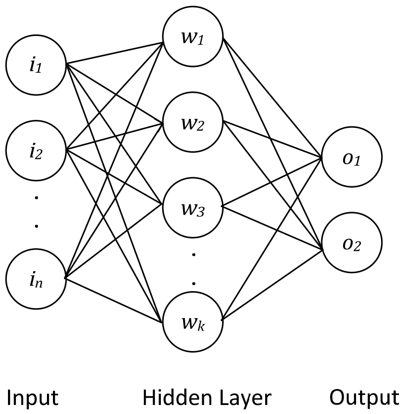


Figure 1. ANN Topology

A. Implementation of Differential search algorithm (DSA)

Usually, in literature Levenberg-Marquardt (LM) [27] optimization algorithm is used and employed to train an ANN with the purpose of determining suitable coefficient values for hidden layers. LM algorithm is one of the most used and robust optimization method which can be applied to any mathematical modeled system in order to optimize while using so little resources and converging to an acceptable result.

Nature-inspired heuristic methods are also studied in literature that some of popular algorithms are used to train ANNs with the aim of obtaining better results. Compared to the LM algorithm, heuristic algorithms require high computation ability and more simulation time to converge best optimal network weights. But the heuristic trained network outputs are much closer to an ideal result which makes the heuristic training methods indispensable.

Differential search (DSA) algorithm is a newly proposed optimization method that simulates the movement behavior of migrating organisms or super-organisms. Pseudo organisms created in DSA imitate the living and migrating organism's behavior in order to find and access the best nutrient sources. This feature helps the organism to walk and move high quality and high variation of nutrient sources. In this algorithm, pseudo-superorganisms migrate to local and global minimums of studied problem.

DSA is similar to other heuristic methods that use randomized initial parameters and system constraints. The pseudo-superorganisms represent the possible solution sets for ANN weights and bias factors which affect networks input and output connection.

In this paper, DSA is applied to ANN with the mostly used optimization algorithm, LM where these separate algorithms optimize network weights for the best system output. Flowchart of the training sequence is given in Figure 2.

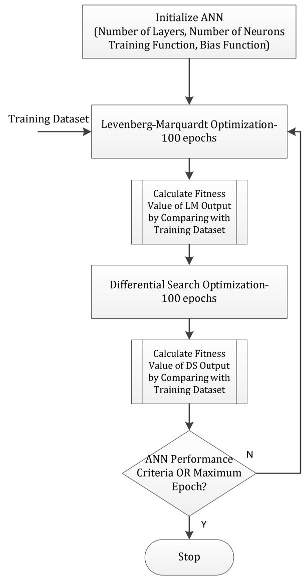


Figure 2. Training process of ANN

IV. Test Results And Discussion

In order verify the efficiency of the proposed hybrid DSA implemented ANN method, three of the most used test systems in literature are chosen which are IEEE 9-bus, IEEE 30-bus and IEEE 118-bus power systems. ANN topology given in Figure 1 is used with parameters and options given in Table I. Being a powerful and robust load flow tool, MATPOWER (MP) [28] add-on for MATLAB is used to train and validate the efficiency of proposed methodology.

TABLE I. NETWORK PARAMETERS

Network	Parameters	Description
Type	Feed-forward back propagation	-
Input Data	Pgen, Vgen, Pload, Qinj	Power generation, generation bus voltages, load demand and injected reactive power
Output Data	Pslack, Gencost, Ploss, Vdev	Slack bus power, total generation cost, total active power loss, voltage deviation of load buses
Training Function	Hybrid	Hybrid training method by using LM & DSA
Adaptation Learning Function	Gradient descent with momentum weight and bias learning	-
Performance Function	MSE	Mean-square error

Number of Layers	2	-
Number of Neurons per Layer	5, 10, 15	Different number of neurons are used and trained
Transfer Function	Tan-Sig	Hyperbolic tangent sigmoid function

A. IEEE 9-bus Test System

IEEE 9-bus system consists of 3 generators and 3 active-reactive loads that the system is studied and tested for various situations in the literature. System data are given by Table II and III, minimum and maximum bus voltages are considered between 0.9 and 1.1 p.u whereas generator constraints are also given in literature [29, 30]. The total demand is 315 MW for active power and 115 MVAR for reactive power. System load, generator and branch data with MP power flow output are given in Table II and Table III for standard parameters studied in the literature.

In order to obtain results for static dataset given in the literature or a randomized dataset, system parameters are processed by using iterative Netwon-Raphson and Gauss-Siedel methods which are implemented in MP add-on.

TABLE II. GENERATOR AND LOAD DATA OF IEEE 9-BUS SYSTEM

Bus	**P_D**	**Q_D**	**P_G**	**Q_G**	**V_{MIN}**	**V_{MAX}**	**V_{BUS}**
1[a]	-	-	71.9574*	27.045*	0.9	1.1	1.040
2[b]	-	-	163	6.6536*	0.9	1.1	1.025
3[b]	-	-	85	-10.8597*	0.9	1.1	1.025
4[c]	-	-	-	-	0.9	1.1	1.025*
5[c]	90	30	-	-	0.9	1.1	1.012*
6[c]	-	-	-	-	0.9	1.1	1.032*
7[c]	100	35	-	-	0.9	1.1	1.015*
8[c]	-	-	-	-	0.9	1.1	1.025*
9[c]	125	50	-	-	0.9	1.1	0.995*

*Values are obtained by using MATPOWER (Newton) iterative power flow method
a:sw bus, b: pv bus, c: pq bus

TABLE III. BRANCH DATA OF IEEE 9-BUS SYSTEM

From-To	**P_{INJ} (MW)**	**Q_{INJ} (MVAr)**	**P_{ABS} (MW)**	**Q_{ABS} (MVAr)**	**P_{LOSS}**	**Q_{LOSS}**
1-4	71.6410	27.0459	-71.6410	-23.9231	0	3.12
4-5	30.7036	1.0300	-30.5372	-16.5433	0.16	0.90
5-6	-59.4627	-13.4566	60.8165	-18.0748	1.35	5.90
3-6	85.0000	-10.8597	-85.0000	14.9553	0	4.10
6-7	24.1834	3.1195	-24.0954	-24.2958	0.08	0.75
7-8	-75.9045	-10.7041	76.3798	-0.7973	0.47	4.03
8-2	-163.00	9.1781	163.0000	6.6536	0	15.83
8-9	86.6201	-8.3808	-84.3201	-11.3127	2.30	11.57
9-4	-40.6798	-38.6872	40.9373	22.8931	0.25	2.19

In order to train the proposed ANN for IEEE 9-bus power system, training dataset including 1000 samples of system parameters is randomized. Training dataset also consists of output data which are obtained by using MP Newton-Raphson based iterative power flow method on randomized system parameters. The input and output datasets are organized as *[P_{LOAD} P_{GEN} V_{GEN}]* which consists of 8 elements and *[P_{SLACK}]* consists of 1 element, respectively. In first case, proposed hybrid ANN is trained by using 5-neurons per layer for maximum epoch of 1000 by using LM and DS hybrid training method. The performance of ANN is obtained by 0.0970 which shows the current topology does not converge to a good result where MSE should be lower than 1e-3. Regression chart for the 5-neurons topology is given in Figure 3. If the regression chart is investigated, it is seen that the performance of test results are similar to validation results since training and test datasets are created randomly. For a scenario of working with real-world parameters or lower range of minimum and maximum limit values, 5-neurons topology may also output desired results with lower MSE.

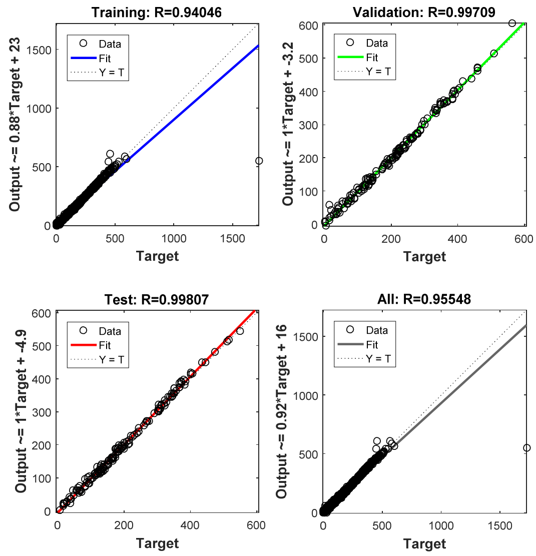
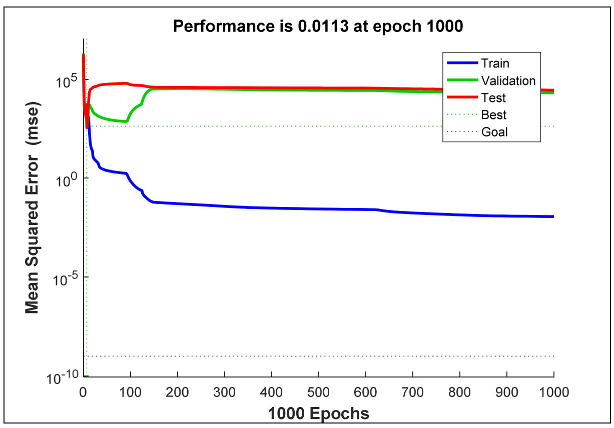


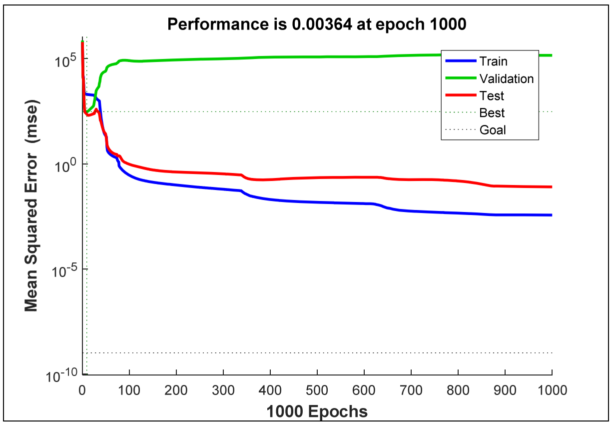
Figure 3. ANN regression chart for IEEE 9-bus test system by using 5-neurons topology

With the purpose of increasing network performance, number of neurons per layer is chosen as 10 and 15 where ANN performance for each topology is obtained as 0.01113 and 0.00364, respectively.

Training state charts for ANN performance are given in Figure 4.a and Figure 4.b. In addition to performance charts, random test data is given input and the results are shown in Table IV which shows that the 15-neurons topology converges to the best result compared to other topologies.



a)



b)

Figure 4. ANN performance for IEEE 9-bus test system a) 10-neurons topology b) 15-neurons topology

TABLE IV. TEST RESULTS FOR IEEE 9-BUS POWER SYSTEM

	Variable	Test-1	Test-2	Test-3	Test-4	Test-5
Input	P_{L1}	143,202	193,360	68,834	119,832	137,271
	P_{L2}	124,427	124,386	14,863	165,520	36,507
	P_{L3}	252,620	204,734	30,669	226,097	157,762
	P_{G2}	173,694	124,590	27,056	233,848	138,941
	P_{G3}	121,836	130,050	50,01	249,987	181,451
	V_{G1}	1,072	0,969	1,021	1,010	0,966
	V_{G2}	0,951	1,078	1,012	1,014	1,017
	V_{G3}	1,015	1,010	1,059	1,000	1,059
Output (P_{Slack})	Newton-Raphson	271,001	324,031	4,255	201,662	115,278
	Gauss-Siedel	271,001	324,031	4,255	201,662	115,278
	5-N Topology	269,471	323,963	6,237	201,882	113,978
	10-N Topology	271,501	323,797	3,686	202,354	115,453
	15-N Topology	270,989	324,044	4,282	201,63	115,260

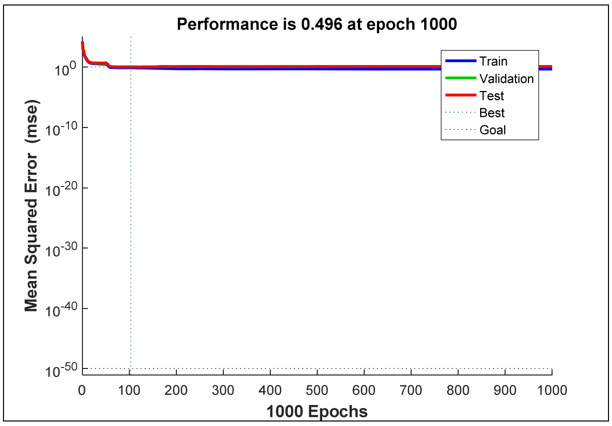
B. IEEE 30-bus Test System

One of the most used test systems, IEEE 30-bus system [31] has been used to show the efficiency of the proposed Hybrid ANN method.

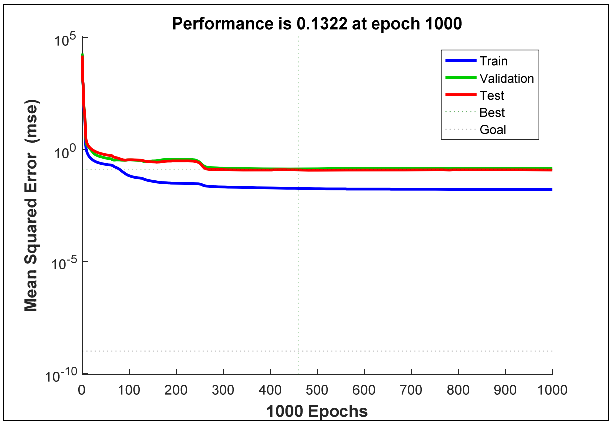
The system has six generators at bus 1, 2, 5, 8, 11, and 13 and four transformers with off-nominal tap ratio at lines 6–9, 6–10, 4–12, and 28–27. In addition, there are shunt VAR compensation devices connected to bus 10, 12, 15, 17, 20, 21, 23, 24, and 29 that can be controlled within the constraints given in literature. Active and reactive power demands of the system are 283.4 MW and 126.2 MVAR, respectively.

System busbar voltage magnitudes are considered between 0.95-1.1 for all buses. In order to train the hybrid ANN for IEEE 30-bus power system, training dataset including 1000 samples is used by MP Newton-Raphson based iterative power flow method.

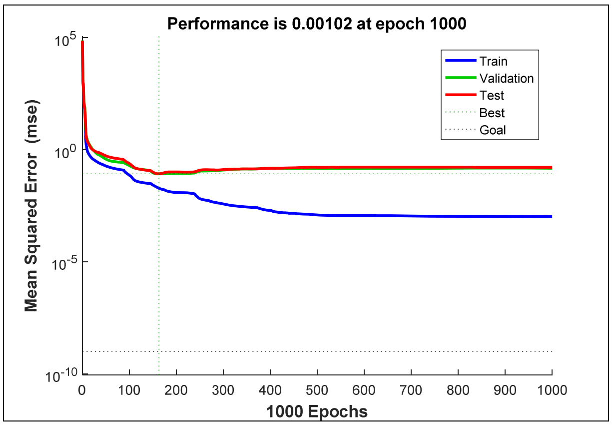
The input and output datasets are organized [P_{LOAD} P_{GEN} V_{GEN} Q_{INJ} T_{LTC}] and [P_{SLACK}] respectively. The performance charts for different topologies and the regression chart for 15-neurons topology are shown in Figure 5 and Figure 6, respectively. If charts are investigated, it is seen that the 15-neurons topology for IEEE 30-bus test system converges to better performance whereas 5 and 10-neurons topologies for IEEE 9-bus test system give better results.



a)



b)



c)

Figure 5. ANN performance for IEEE 30-bus test system a) 5-neurons topology b) 10-neurons topology c) 15-neurons topology

As the system get complex and have more dependent variables, ANN topology need more neurons per layer in order to converge the system in given constraints. Test data is used in order to show effectiveness of proposed hybrid ANN method, results are shown in Table V.

TABLE V. TEST RESULTS FOR IEEE 30-BUS POWER SYSTEM

	Variable	Test-1	Test-2	Test-3	Test-4	Test-5
Input	P_{G2}	295.814	169.720	652.084	21.377	52.309
	P_{G3}	43.544	167.526	30.394	419.346	435.836
	P_{G4}	419.033	22.923	112.348	281.345	216.087
	P_{G5}	149.082	282.356	47.651	24.623	156.822
	P_{G6}	351.788	40.050	161.931	115.404	64.156
	V_{G1}	1.0440	1.0939	1.0091	0.9403	0.9728
	V_{G2}	1.0473	1.0362	1.0908	0.9162	1.0025
	V_{G3}	0.9947	0.9932	0.9468	1. 0627	0.9674

	V_{G4}	0.9678	1.0051	1.0071	0.9190	1.0115
	V_{G5}	0.9020	0.9230	0.9939	1.0748	0.9142
	V_{G6}	0.9657	1.0350	0.9295	1.0825	1. 0412
Output (P_{Slack})*	Newton-Raphson	295.814	169.720	652.084	21.377	52.309
	Gauss-Siedel	295.815	169.720	652.084	21.379	52.309
	5-N Topology	298.421	171.472	650.418	20.893	53.993
	10-N Topology	294.226	169.130	653.717	22.953	53.522
	15-N Topology	295.621	170.513	652.230	21.393	51.877

*Power flow is run with different load demands within the constraints; load data could not be included in table due to being large in number.

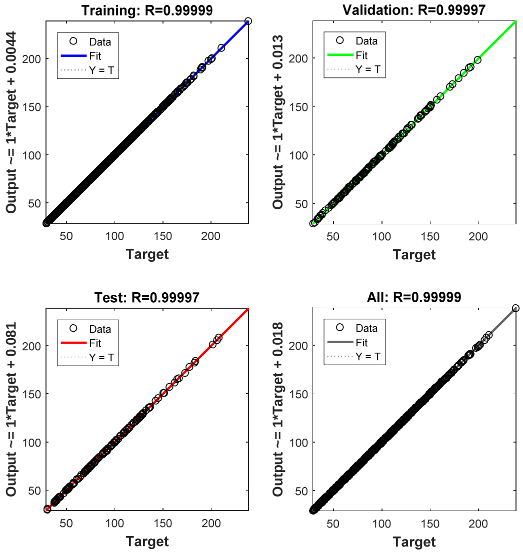


Figure 6. ANN regression for IEEE 30-bus test system by using 15-neurons topology

C. IEEE 118-bus Test System

IEEE 118-bus system [32] has fifty-four generators and nine transformers with off-nominal tap ratio. There are also thirteen shunt VAR compensation devices connected different buses on the system given in literature constraints. System busbar voltage magnitudes are considered between 0.95-1.1 p.u. for all buses whereas the tap settings and shunt VAR device limits are taken as 0.9-1.1 and 0-0.4 p.u., respectively.

The default total active power demand of the system is 4242 MW as given by the literature. System is also used in order to verify and show effectiveness of proposed hybrid ANN method in power flow problems where training datasets are organized as [PLOAD PGEN VGEN QINJ TLTC] and [PSLACK].

Due to the complexity and large number of parameters ANN performance was able to converge to the best result of 201e-3 which is not a satisfying value. Training results for 5, 10 and 15 neurons topologies are given by Figure 7. In order to improve the network response and show efficiency of proposed method, 30-neurons topology is used; performance and regression charts are shown in Figure 8 and Figure 9, respectively.

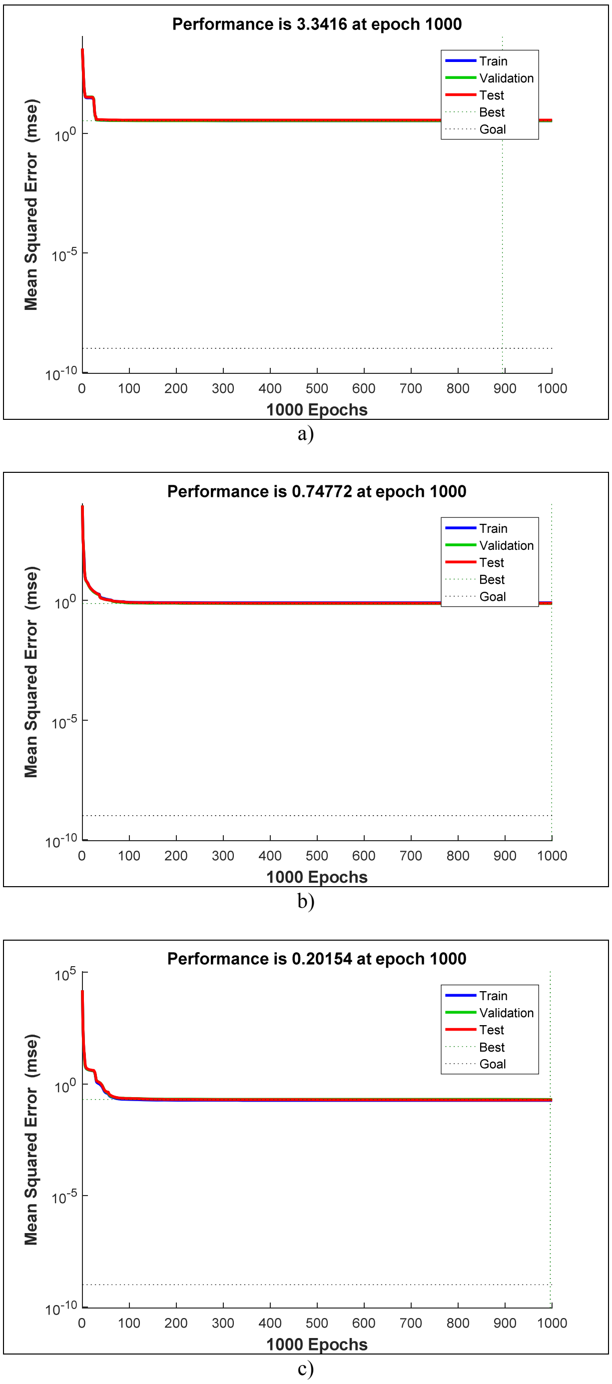


Figure 7. ANN performance for IEEE 118-bus test system a) 5-neurons topology b) 10-neurons topology c) 15-neurons topology

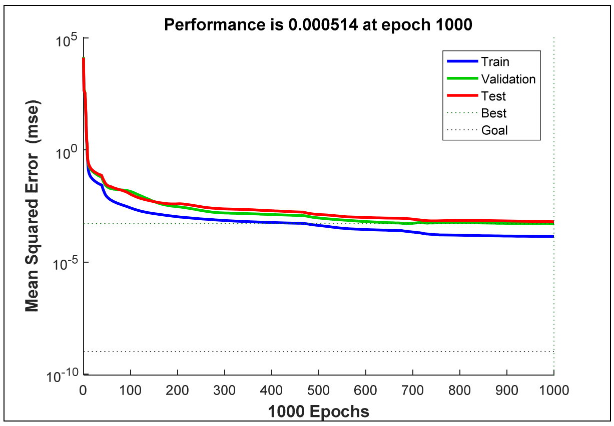


Figure 8. ANN performance for IEEE 118-bus test system by using 30-neurons topology

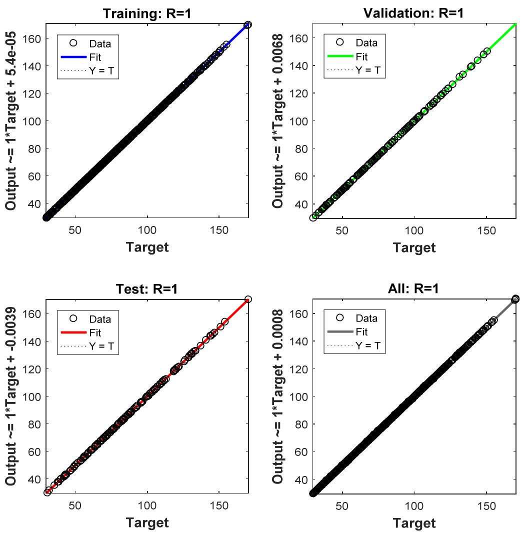


Figure 9. ANN regression for IEEE 118-bus test system by using 30-neurons topology

If Figure 8 is investigated, it is seen that the best result is achieved for the epoch close to 1000, which shows that 30-neurons topology may also give better results compared to 5, 10 and 15-neurons topologies with higher of epoch number.

As seen by the results, increasing neurons per layer boosts the network performance. On the other hand, in order to compute and obtain the best network parameters by running simulations consumes great amount of time since a heuristic based hybrid approach is used and applied to the system. The overall data of simulation process for 9, 30 and 118-bus test systems is given by Table IV. Simulations for each number of neurons and power systems are run 100 times in order to determine average training time where the time difference, performance for test systems and neurons per layers can be seen clearly.

TABLE VI. TRAINING PROCESS DETAILS FOR HYBRID-ANN METHOD

Sys.	Number of Neurons	Levenberg-Marquardt Training		Hybrid ANN Training	
		Best Perform.	Training Time*	Best Perform.	Training Time*
9-Bus	5-N	2.31	3.92	97e-3	13.1
	10-N	1.98	8.12	11.3e-3	32.7
	15-N	0.671	13.3	3.64e-3	42.1
30-Bus	5-N	6.81	24.8	0.496	62.4
	10-N	3.22	52.5	0.132	101.5
	15-N	0.946	91.4	1.02e-3	144.9
118-Bus	5-N	149.2	382.2	3.341	819.7
	10-N	81.5	927.4	0.747	3023.4
	15-N	59.9	1484.5	0.201	5074.1
	30-N	1.33	8e3	0.5e-3	18e3

*Average training time in seconds.

V. CONCLUSION

In this work, the power flow problem is successfully solved by using the hybrid trained ANN. The proposed hybrid training algorithm is studied on different ANN topologies in order to converge to the actual output with the best result while considering computation time and randomized training dataset number. This work is focused on achieving the power flow of IEEE 9, IEEE 30 and IEEE 118 bus test systems whose initial parameters are randomized in respect of constraints. The obtained results are compared with the widely used iterative methods, Newton-Raphson and Gauss-Siedel methods, whose efficiency and output accuracy are well known. It is seen that the proposed hybrid ANN approach can be used in order to solve complex and dynamic power flow problems. In addition to this, after training the network successfully, power flow problems can easily be solved without need of any specific software by using determined ANN parameters. Thus, the proposed ANN method can be used in order to achieve dynamic power flow optimization, control and fault prediction without requiring long duration simulations and high processing load.

REFERENCES

[1] K. Abaci, V. Yamacli, "Optimal Reactive-Power Dispatch Using Differential Search Algorithm," Electrical Engineering, vol. 99(1), pp. 213-225, 2017. doi: 10.1007/s00202-016-0410-5
[2] J.A. Momoh, J.Z. Zhu, "Improved Interior Point Method For OPF Problems," IEEE Trans. on Power Systems, vol. 14(3), pp. 1114-1120, 1999. doi: 10.1109/59.780938
[3] J. Carpentier, "Contribution a l'etude du Dispatching Economique," Bulletin de la Societe Francaise des Electriciens, vol. 3, pp.431-474, 1962.
[4] A.A. Abou El, M. A. Abido, "Optimal Power Flow Using Tabu Search Algorithm," Electric Power Components and Systems, vol. 30(5), pp. 469-483, 2002. doi: 10.1080/15325000252888425
[5] R. Mota-Palomino, V.H. Quintana, "Sparse Reactive Power Scheduling By A Penalty-Function Linear Programming Technique," IEEE Transactions on Power Systems, vol. 1(3), pp. 31-39, 1986. doi: 10.1109/TPWRS.1986.4334951
[6] R.C. Burchett, H. Happ, D. R. Vierath, "Quadratically Convergent Optimal Power Flow," IEEE Trans. on Power Appar. Syst., vol. 103(11), pp. 3264-3276, 1984. doi: 10.1109/TPAS.1984.318568
[7] D.I. Sun, B. Ashley, B. Brewer, A. Hughes, "Optimal Power Flow by Newton Approach," IEEE Trans. Power Appar. Syst., vol. 103(10), pp. 2864-2875, 1984. doi: 10.1109/TPAS.1984.318284
[8] A. Santos, Jr. da Costa, "Optimal Power Flow Solution By Newton's Method Applied To An Augmented Lagrangian Function," IEEE Proc. Gener. Transm. Distrib. Vol 142(1), pp. 33-36, 1995. doi: 10.1049/ip-gtd:19951586
[9] M. Rahli, P. Pirotte, "Optimal Load Flow Using Sequential Unconstrained Minimization Technique Method Under Power Transmission Losses Minimization," Electrical Power System Research, vol 52, pp. 61-64, 1999. doi: 10.1016/S0378-7796(99)00008-5
[10] X. Yan, V.H. Quintana, "Improving An Interior Point Based OPF By Dynamic Adjustments of Step Sizes and Tolerances," IEEE Trans. Power Syst., vol. 14(2), pp. 709-717, 1999. doi: 10.1109/59.761902
[11] V. Veerasamy, R. Ramachandran, M. Thirumeni, B. Madasamy, "Load Flow Analysis Using Generalised Hopfield Neural Network," IET Generation, Transmission & Distribution, vol. 12, pp. 1765-1773, 2017. doi: 10.1049/iet-gtd.2017.1211
[12] D. Karaboga, B. Akay, "Artificial Bee Colony (ABC) Algorithm on Training Artificial Neural Networks," in Proc. 15th Signal Processing and Communications Applications, Eskisehir, 2007, pp. 128-139.
[13] D. Karaboga, C. Ozturk, "Neural Networks Training By Artificial Bee Colony Algorithm on Pattern Classification," Neural Network World, vol. 19, pp. 279-292, 2009.
[14] J.A. Bullinaria, K. AlYahya, "Artificial Bee Colony Training of Neural Networks: Comparison With Back-Propagation" Memetic Comp. vol. 6, pp. 6-171, 2014. doi: 10.1007/s12293-014-0137-7
[15] T. Olsson, K. Magnusson, "Training artificial neural networks with genetic algorithms for stock forecasting" KTH Royal Institute Of Technology School Of Computer Science And Communication, Stockholm, 2016, vol. 1.
[16] L. Hu, L Q, K. Mao, W. Chen, X. Fu, "Optimization of Neural Network By Genetic Algorithm for Flowrate Determination In Multipath Ultrasonic Gas Flowmeter," IEEE Sensors Journal, vol. 16(5), pp. 1158-1167, 2016. doi: 10.1109/JSEN.2015.2501427
[17] F. Valdez, O. Castillo, P. Melin, "Ant Colony Optimization for the Design Of Modular Neural Networks in Pattern Recognition," in Proc. International Joint Conference on Neural Networks, Canada, 2016, pp. 24-29. doi: 10.1109/IJCNN.2016.7727194

[18] C. Juang, Y. Yeh, “Multiobjective Evolution of Biped Robot Gaits Using Advanced Continuous Ant-Colony Optimized Recurrent Neural Networks,” IEEE Trans. on Cybernetics, vol. 48(6), pp. 1910-1922, 2018. doi: 10.1109/TCYB.2017.2718037

[19] P. Civicioglu, “Transforming Geocentric Cartesian Coordinates to Geodetic Coordinates by Using Differential Search Algorithm,” Computers & Geosciences, vol. 46, pp. 229-247, 2012. doi: 10.1016/j.cageo.2011.12.011

[20] K. Abaci, V. Yamacli, "Differential Search Algorithm for Solving Multi-Objective Optimal Power Flow Problem," International Journal of Electrical Power & Energy Systems, vol. 79, pp. 1-10, 2016. doi: 10.1016/j.ijepes.2015.12.021

[21] T. Kurban, P. Civicioglu, R. Kurban, E. Besdok “Comparison of Evolutionary and Swarm Based Computational Techniques for Multilevel Color Image Thresholding,” Applied Soft Computing, vol. 23(1), pp. 128-143, 2014. doi: 10.1016/j.asoc.2014.05.037

[22] M. A. Gunen, P. Civicioglu, E. Besdok “Differential Search Algorithm Based Edge Detection,” in Proc. International Society for Photogrammetry and Remote Sensing Congress, Chezch Republic, 2016, pp. 667-670.

[23] H. M. Marghny, R. M. A. Elaziz, A. I. T. Mohamed, “Differential Search Algorithm-based Parametric Optimization of Fuzzy Generalized Eigenvalue Proximal Support Vector Machine,” International Journal of Computer Applications, vol. 108(19), pp. 38-46, 2014. doi: 10.5120/19023-0540

[24] H. Beirami, A. Z. Shabestari, M. M. Zerafat, “Optimal PID Plus Fuzzy Controller Design for a PEM Fuel Cell Air Feed System Using The Self-Adaptive Differential Evolution Algorithm,” International Journal of Hydrogen Energy, vol. 40(30), pp. 9422-9434, 2015. doi: 10.1016/j.ijhydene.2015.05.114

[25] F. Amato, A. Lopez-Rodriguez, E.M. Pena-Mendez, P. Vanhara, “Artificial Neural Networks in Medical Diagnosis,” Journal of Applied Biomedicine, vol. 11(2), pp. 47-58, 2013. doi: 10.2478/v10136-012-0031-x

[26] Z. Comert, A.F. Kocamaz, “A Study of Artificial Neural Network Training Algorithms for Classification of Cardiotocography Signals,” Journal of Science and Technology, vol. 7(2), pp. 93-103, 2017.

[27] C. Ozturk, D. Karaboga, “Hybrid Artificial Bee Colony Algorithm for Neural Network Training,” in Proc. IEEE Congress of Evolutionary Computation, USA, 2011, pp. 84-88.

[28] R.D. Zimmerman, C.E. Murillo-S´anchez, R.J. Thomas, “Matpower: Steadystate Operations, Planning and Analysis Tools for Power Systems Research and Education,” IEEE Trans. Power Syst, vol. 26(1), pp. 12–19, 2011. doi: 10.1109/TPWRS.2010.2051168

[29] P.W. Sauer, M.A. Pai. “Power System Dynamics and Stability”, pp. 218-293, Prentice Hall, 1998.

[30] T.B. Nguyen, M.A. Pai, “Dynamic Security-Constrained Rescheduling of Power Systems Using Trajectory Sensitivities,” IEEE Trans. Power Syst., vol. 18(2), pp. 848-54, 2003. doi: 10.1109/TPWRS.2003.811002

[31] O. Alsac, B. Sttot, “Optimal Load Flow with Steady State Security,” IEEE Trans. Power Appar. Syst., vol. 93, pp. 745-751, 1974. doi: 10.1109/TPAS.1974.293972

[32] I. Pena, C.B. Martinez-Anido, B.M. Hodge, “An Extended IEEE 118-Bus Test System with High Renewable Penetration,” IEEE Trans. Power Syst., vol. 33(1), pp. 281-289, 2018. doi: 10.1109/TPWRS.2017.2695963